

Laboratory for Physical Sciences (LPS) Advanced Computing Systems (ACS)  
Technical Update Seminar  
College Park, Maryland, USA, May 10th, 2023



# The Asynchronous Ballistic Approach to Reversible Computing in Superconductors



Wednesday, May 10<sup>th</sup>, 2023

Michael P. Frank, Center for Computing Research

*with Rupert Lewis (Quantum Phenomena Dept.)*

Approved for public release, SAND2023-03504 PE



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

# Contributors to our Reversible Computing research program



## Full group of recent staff at Sandia:

- Michael Frank (Cognitive & Emerging Computing)
- Robert Brocato (RF Microsystems) – now retired
- David Henry (MESA Hetero-Integration)
- Rupert Lewis (Quantum Phenomena)
  - Terence “Terry” Michael Bretz-Sullivan
- Nancy Missert (Nanoscale Sciences) – now retired
  - Matt Wolak (now at Northrop-Grumman)
- Brian Tierney (Rad Hard CMOS Technology)

## Thanks are also due to the following colleagues & external research collaborators:

- Karpur Shukla (CMU → Flame U. → Brown U.)
  - Currently in Prof. Jimmy Xu’s Lab for Emerging Techs.
- Hannah Earley (Cambridge U. → startup)
- Erik DeBenedictis (Sandia → Zettaflops, LLC)
- Joseph Friedman (UT Dallas)
  - with A. Edwards, X. Hu, B.W. Walker, F. Garcia-Sanchez, P. Zhou, J.A.C. Incorviaz, A. Paler
- Kevin Osborn (LPS/JQI)
  - Liuqi Yu, Ryan Clarke, Han Cai
- Steve Kaplan (independent contractor)
- Rudro Biswas (Purdue)
  - Dewan Woods & Rishabh Khare
- Tom Conte (Georgia Tech/CRNCH)
  - Anirudh Jain, Gibran Essa, Austin Adams
- David Guéry-Odelin (Toulouse U.)
- FAMU-FSU College of Engineering:
  - Sastry Pamidi (ECE Chair) & Jerris Hooker (Instructor)
  - 2019-20 students:
    - Frank Allen, Oscar L. Corces, James Hardy, Fadi Matloob
  - 2020-21 students:
    - Marshal Nachreiner, Samuel Perlman, Donovan Sharp, Jesus Sosa

Thanks are due to Sandia’s LDRD program, DOE’s ASC program, and the DoD/ARO ACI (Advanced Computing Initiative) for their support of this line of research!

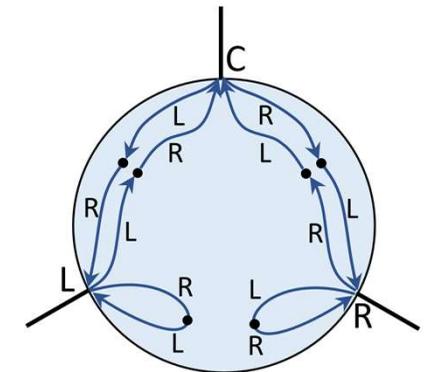
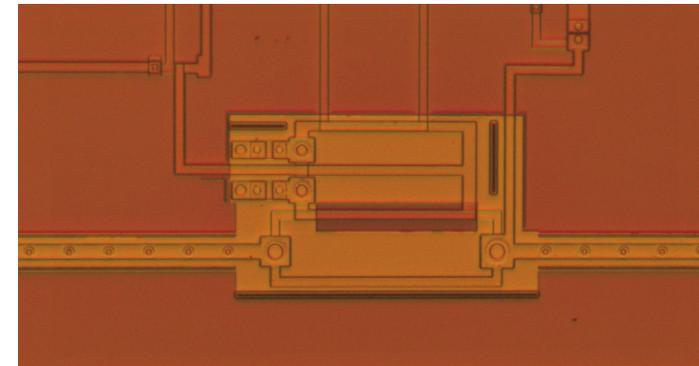
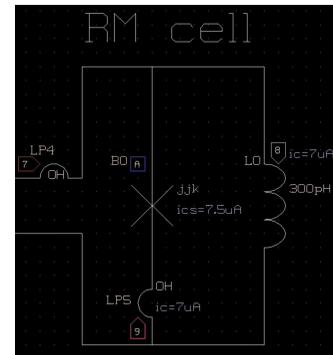
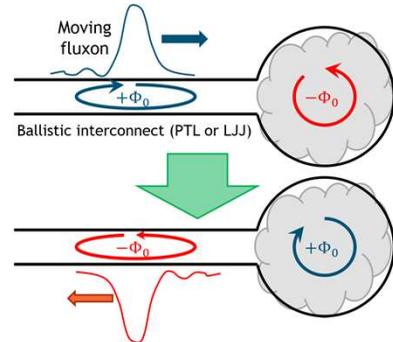
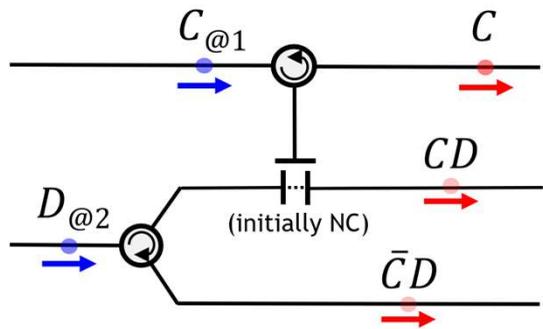
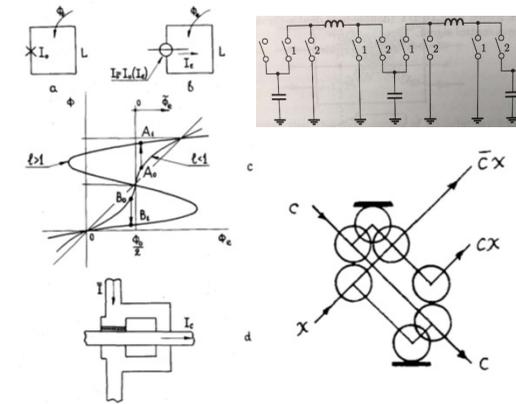
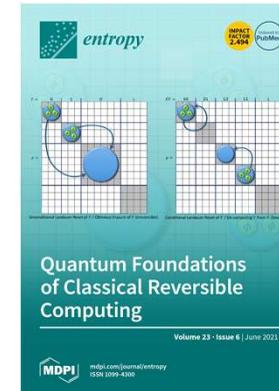
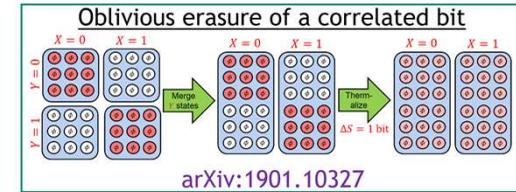
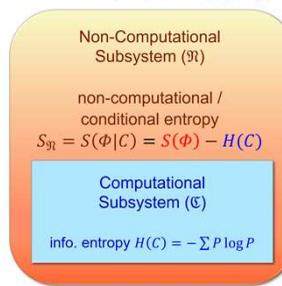


# Talk Abstract/Outline

## Ballistic Asynchronous Reversible Computing in Superconducting Circuits

- Background: *Why Reversible Computing?*
  - Relevant classic results in the thermodynamics of computing
    - Recently generalized to quantum case
  - Two major types of approaches to reversible computing in superconducting circuits:
    - *Adiabatic approaches* – Well-developed today.
      - Likharev's *parametric qnantron* (1977); more recent QFP tech (YNU & collabs.) w. substantial demo chips.
    - *Ballistic approaches* – Much less mature to date.
      - Fredkin & Toffoli's early concepts (1978-'81); much more recent work at U. Maryland, Sandia, UC Davis
- **Review:** The relatively new *asynchronous* ballistic approach to RC in SCE.
  - Addresses concerns w instability of the synchronous ballistic approach
  - Potential advantages of asynchronous ballistic RC (vs. adiabatic approaches)
  - Implementation w. superconducting circuits (BARCS effort).
- **Focus of this Talk:**
  - Presenting our recent work on enumerating/classifying possible BARCS functions w.  $\leq 3$  ports and  $\leq 2$  states.

Computing System ( $\mathcal{C}$ ),  
total entropy  $S(\Phi) = -\sum p \log p$



# Why Reversible Computing?

## Thermodynamics of computing: Relevant classic results

Based on the pioneering historical insights of Landauer & Bennett...

### 1. Fundamental Theorem of the Thermodynamics of Computing →

- Unification of physical and information-theoretic entropy.
  - Implies interconvertibility of computational and non-computational entropy.

### 2. Landauer's Principle (proper) →

- Loss of known/correlated computational information to a thermal environment transforms it into *new* physical entropy.

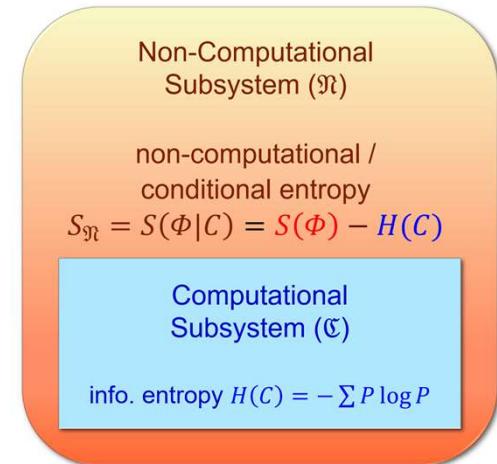
### 3. Conventional digital architectures (which discard correlated information all the time) have a *fundamental* efficiency limit...

- $\geq kT \ln 2$  energy dissipation per bit of information loss.
  - Actual losses per bit erased in practical designs tend to be at least 10s–1000s of  $kT$ .

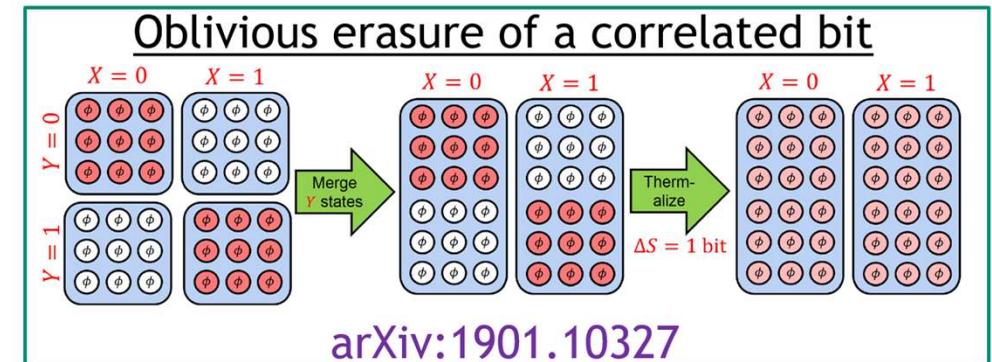
### 4. Alternative *reversible* digital architectures which transform states 1:1 can (at least in principle) avoid the Landauer limit.

- There is no known fundamental efficiency limit for reversible machines.

Computing System ( $\mathfrak{C}$ ),  
total entropy  $S(\Phi) = -\sum p \log p$

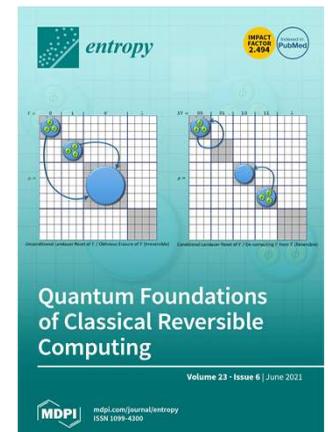
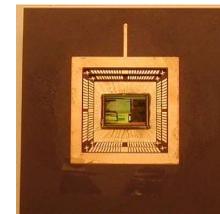


$$P(\mathbf{c}_j) = \sum_{\phi_i \in \mathbf{c}_j} p(\phi_i)$$



Quantum generalizations of classic results surveyed in M. Frank & K. Shukla, doi:10.3390/e23060701 –

Pendulum adiabatic processor (MIT '99)



# The two major approaches to reversible computing

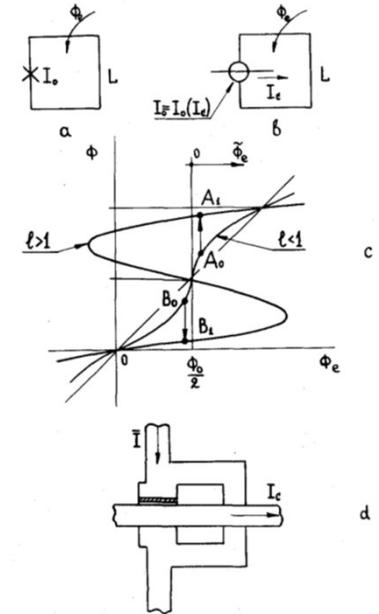
## Both relevant in superconducting electronics

[10.1109/TMAG.1977.1059351](https://doi.org/10.1109/TMAG.1977.1059351):



**Adiabatic approaches** – based on *gradually* transforming a device's potential energy surface

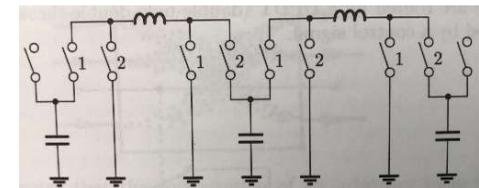
- General method suggested in Landauer's original (1961) paper.
  - By definition, transitions are *slow* compared to the natural relaxation timescale of the device.
- First historical example of an engineered fully adiabatic electronic logic cell:
  - Likharev's *parametric quantron* (1977) – Use a *control current*  $I_c$  to raise/lower the potential energy barrier between loop states.
- Modern AQFP/RQFP technology from YNU has a similar spirit, but is much more well-developed.



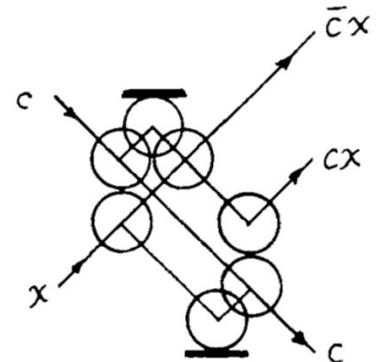
**Ballistic approaches** – based on *ballistic* dynamics & *elastic* interactions between DOFs

- Assumes relatively slight coupling between dynamical DOFs and the thermal environment...
  - Interactions happen *fast* relative to that coupling, so there isn't *time* for the dynamical excited state to relax thermally – dynamical energy largely *conserved* in the DOFs of interest.
- Early electronic & mechanical concepts proposed by Fredkin & Toffoli:
  - Early electronic concept (1978) as an underdamped LC circuit with idealized switches...
  - Simple mechanical thought experiment (1981)... “Billiard Ball Model”
- But, almost no engineering development of this approach from 1980 – 2010!
  - Why? The original concept appeared to have intractable issues w. synchronization / chaotic instabilities...

[10.1007/978-1-4471-0129-1\\_2](https://doi.org/10.1007/978-1-4471-0129-1_2):



[10.1007/BF01857727](https://doi.org/10.1007/BF01857727):



# Ballistic Reversible Computing

Can we envision reversible computing as a deterministic elastic interaction process?

Historical origin of this concept:

- Fredkin & Toffoli's *Billiard Ball Model* of computation ("Conservative Logic," IJTP 1982).
  - Based on elastic collisions between moving objects.
  - Spawned a subfield of "collision-based computing."
    - Using localized pulses/solitons in various media.

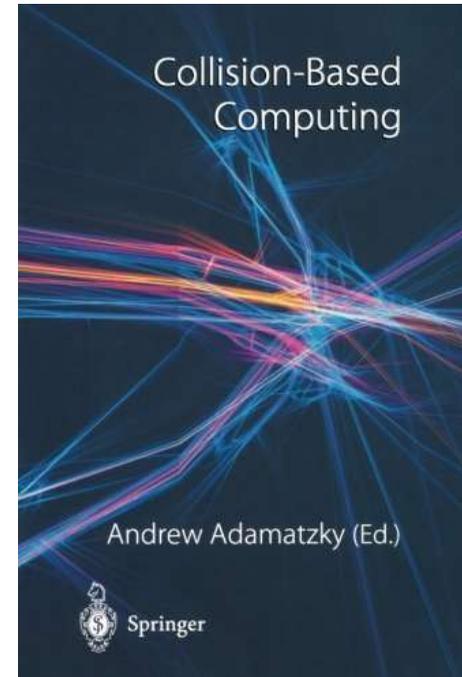
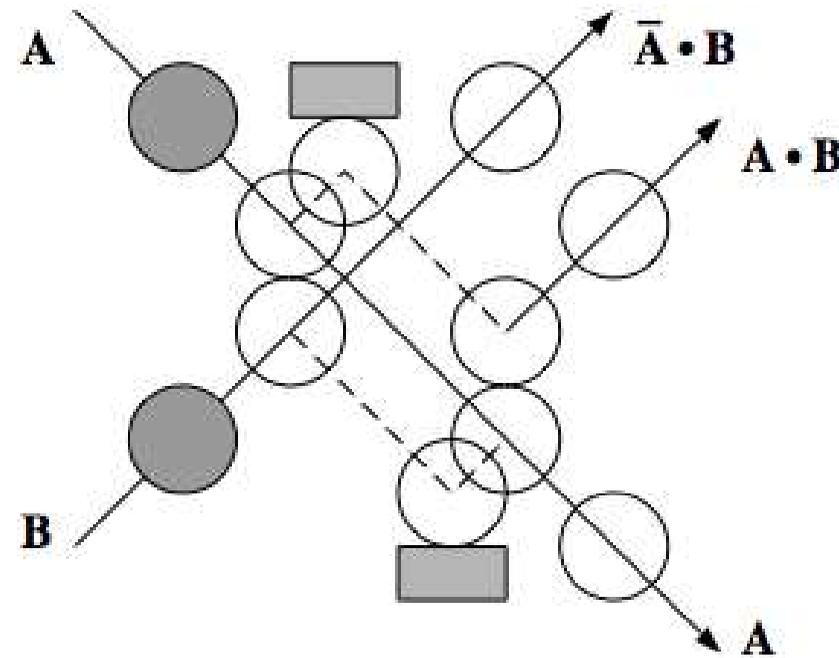
No power-clock driving signals needed!

- Devices operate when data signals arrive.
- The operation energy is carried by the signal itself.
  - (Almost) all of the signal energy is preserved in outgoing signals.

However, all (or almost all) of the existing design concepts for ballistic computing invoke implicitly *synchronized* arrivals of ballistically-propagating signals...

- Making this work in reality presents some serious difficulties, however:
  - Unrealistic in practice to assume precise alignment of signal arrival times.
    - Thermal fluctuations & quantum uncertainty, at minimum, are always present.
  - Any relative timing uncertainty leads to chaotic dynamics when signals interact.
    - $\therefore$  Exponentially-increasing uncertainties in the dynamical trajectory.
  - Deliberate *resynchronization* of signals whose timing relationship is uncertain incurs an inevitable energy cost.

Can we come up with a new ballistic model that avoids or mitigates these problems?



# Ballistic Asynchronous Reversible Computing (BARC)



**Problem:** Conservative (dissipationless) dynamical systems generally tend to exhibit chaotic behavior...

- This results from direct nonlinear *interactions* between multiple continuous dynamical degrees of freedom (DOFs), which amplify uncertainties, exponentially compounding them over time...
- *E.g.*, positions/velocities of ballistically-propagating “balls”
  - Or more generally, any localized, cohesive, momentum-bearing entity: Particles, pulses, quasiparticles, solitons...

**Core insight:** In principle, we can greatly reduce or eliminate this tendency towards dynamical chaos...

- We can do this simply by *avoiding* any direct interaction between continuous DOFs of different ballistically-propagating entities

Require localized pulses to arrive *asynchronously*—and furthermore, at clearly distinct, *non-overlapping* times

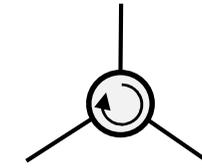
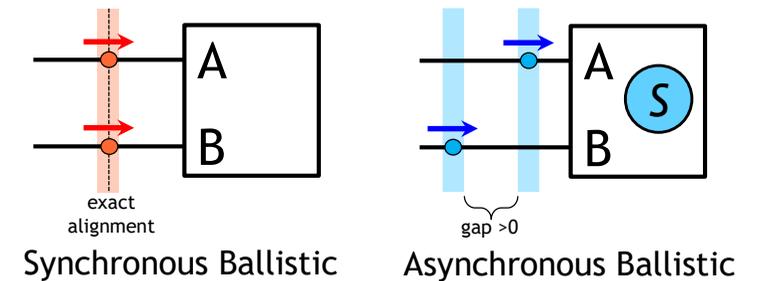
- Device’s dynamical trajectory then becomes *independent* of the precise (absolute *and* relative) pulse arrival times
  - As a result, timing uncertainty per logic stage can now accumulate only *linearly*, not exponentially!
    - Only relatively occasional re-synchronization will be needed
- For devices to still be capable of doing logic, they must now maintain an internal discrete (digitally-precise) state variable—a stable (or at least metastable) stationary state, *e.g.*, a ground state of a well

No power-clock signals, unlike in adiabatic designs!

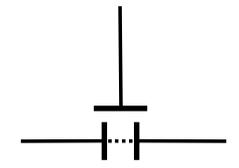
- Devices simply operate whenever data pulses arrive
- The operation energy is carried by the pulse itself
  - Most of the energy is preserved in outgoing pulses
    - Signal restoration can be carried out incrementally

**Goal of current effort at Sandia:** Demonstrate BARC principles in an implementation based on fluxon dynamics in Superconducting Electronics (SCE)

(BARCS  effort)

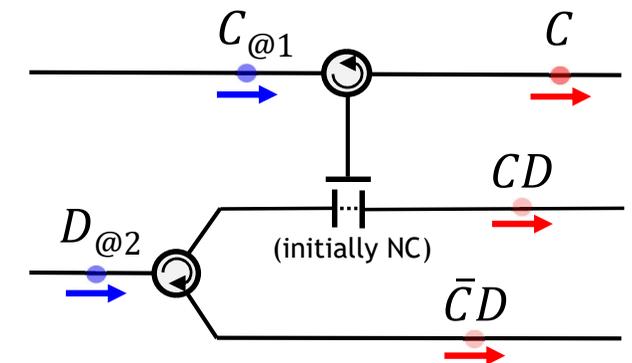


Rotary  
(Circulator)



Toggled  
Barrier

Example BARC device functions



Example logic construction

# Simplest Fluxon-Based (bipolarized) BARC Function

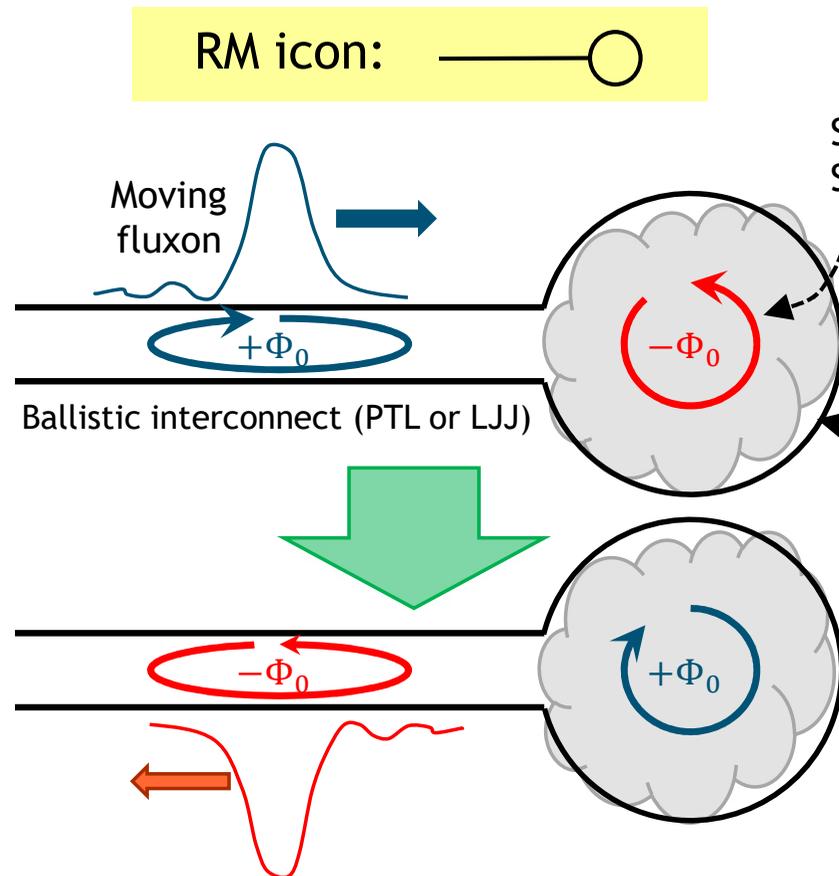


One of our early tasks: Characterize the simplest nontrivial BARC device functionalities, given a few simple design constraints applying to an SCE-based implementation, such as:

- (1) Bits encoded in fluxon polarity; (2) Bounded planar circuit conserving flux; (3) Physical symmetry.

Determined through theoretical hand-analysis that the simplest such function is the ***1-Bit, 1-Port Reversible Memory Cell (RM)***:

- Due to its simplicity, this was then the preferred target for our subsequent detailed circuit design efforts...



Stationary SFQ

Some planar, unbiased, reactive SCE circuit w. a continuous superconducting boundary

- Only contains L's, M's, C's, and *unshunted* JJs
- Junctions should mostly be *subcritical* (avoids  $R_N$ )
- Conserves total flux, approximately nondissipative

Desired circuit behavior (NOTE: conserves flux, respects T symmetry & logical reversibility):

- If polarities are opposite, they are swapped (shown)
- If polarities are identical, input fluxon reflects back out with no change in polarity (not shown)
- (*Deterministic*) *elastic 'scattering'* type interaction: Input fluxon kinetic energy is (nearly) preserved in output fluxon

RM Transition Table

Input Syndrome		Output Syndrome
+1(+1)	→	(+1)+1
+1(-1)	→	(+1)-1
-1(+1)	→	(-1)+1
-1(-1)	→	(-1)-1

# RM—First working (in simulation) implementation!



Erik DeBenedictis: “Try just strapping a JJ across that loop.”

- This actually works!

“Entrance” JJ sized to = about 5 LJJ unit cells ( $\sim 1/2$  pulse width)

- I first tried it twice as large, & the fluxons annihilated instead...
  - “If a  $15 \mu\text{A}$  JJ rotates by  $2\pi$ , maybe  $1/2$  that will rotate by  $4\pi$ ” 🤔

Loop inductor sized so  $\pm 1$  SFQ will fit in the loop (but not  $\pm 2$ )

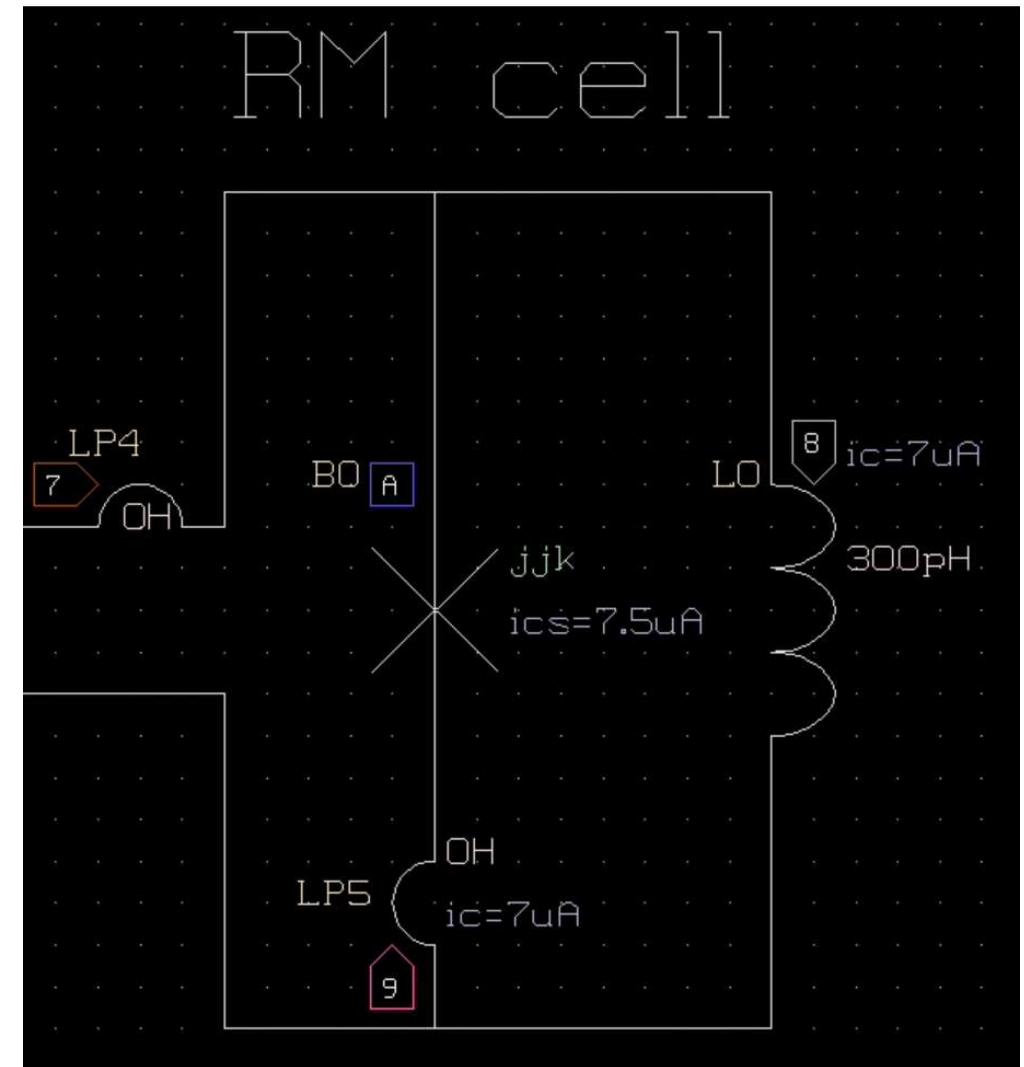
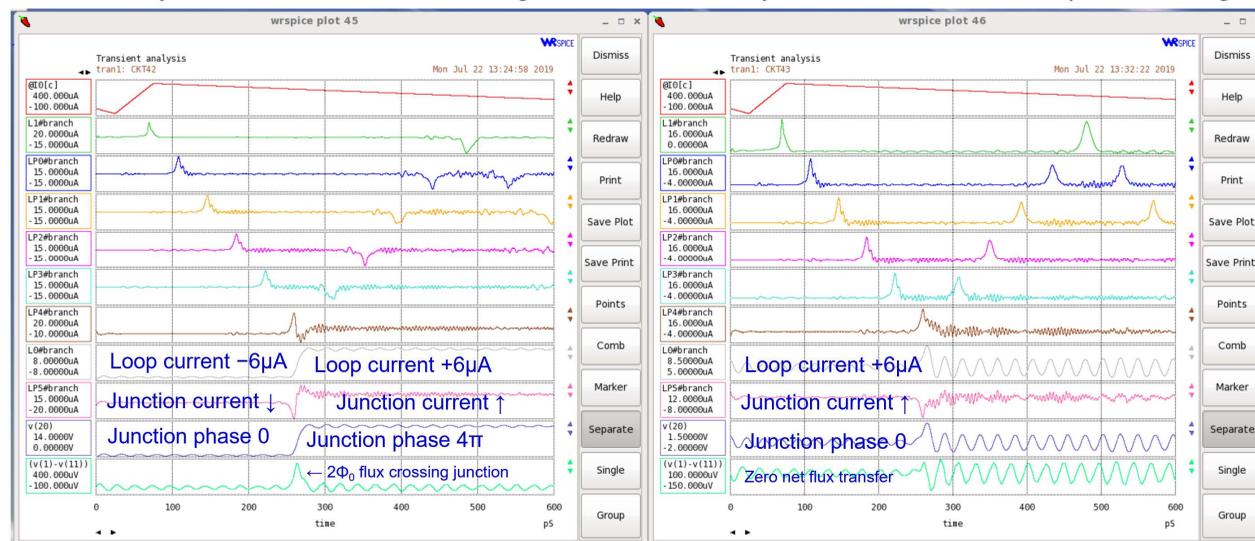
- JJ is sitting a bit below critical with  $\pm 1$

WRspice simulations with  $\pm 1$  fluxon initially in the loop

- Uses `ic` parameter, & `uic` option to `.tran` command
  - Produces initial ringing due to overly-constricted initial flux
  - Can damp w. small shunt  $G$

Polarity mismatch  $\rightarrow$  Exchange

Polarity match  $\rightarrow$  Reflect (=Exchange)

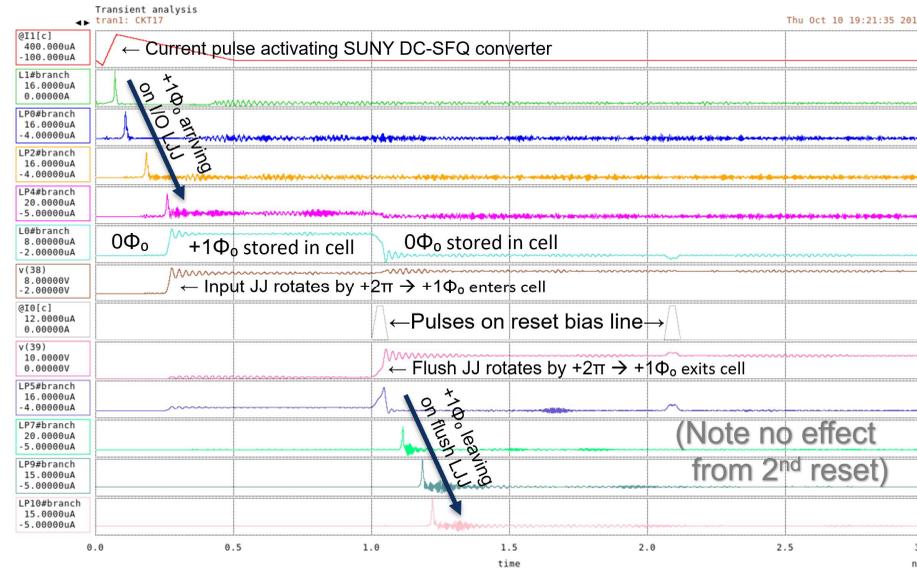
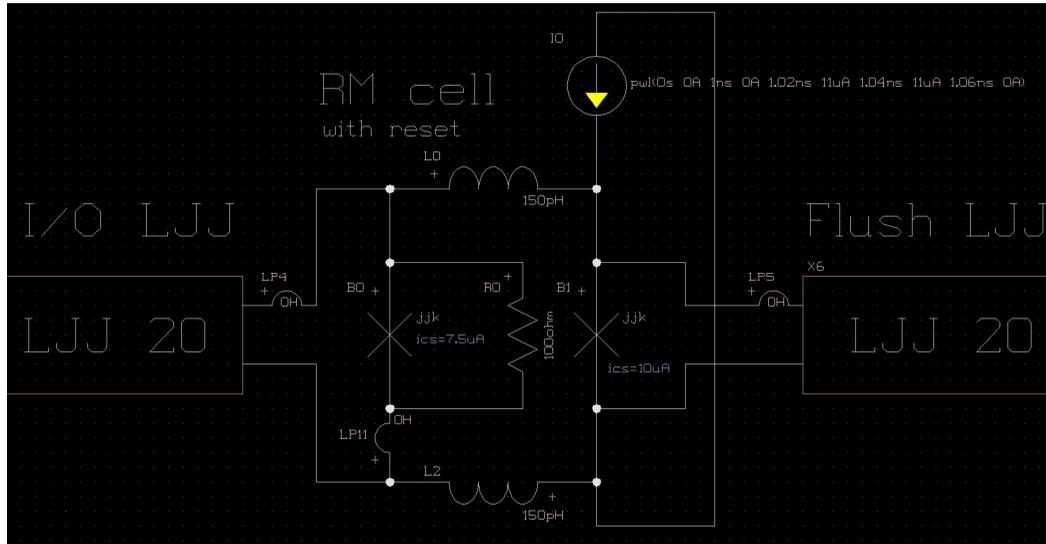


# Resettable version of RM cell—Designed & Fabricated!



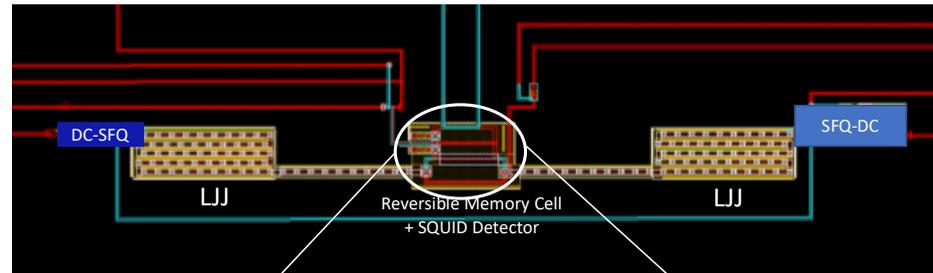
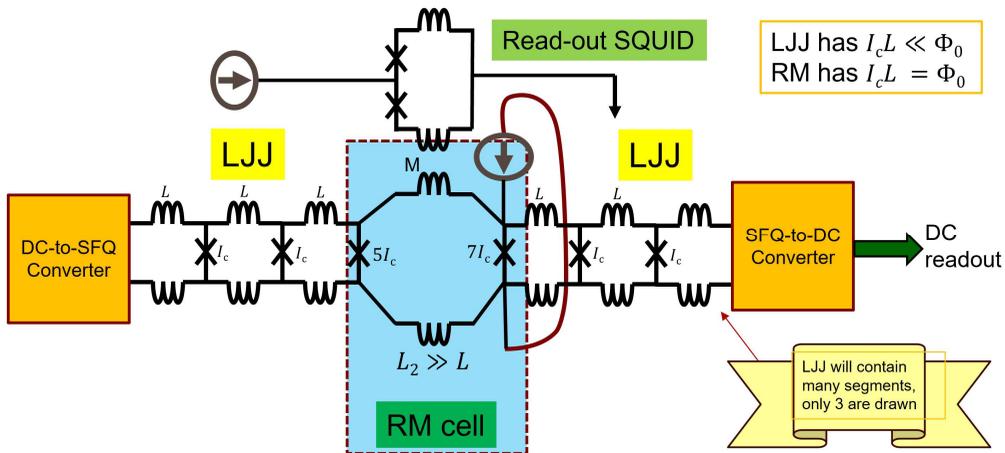
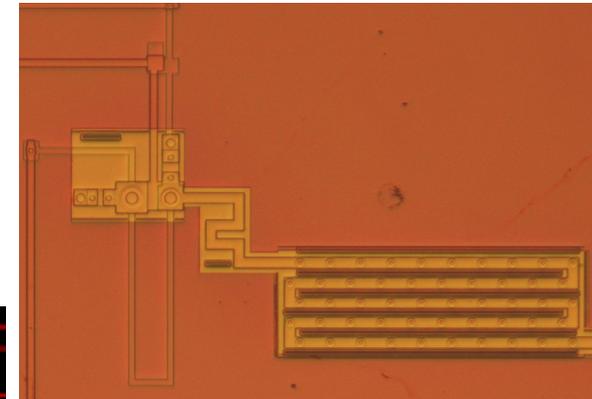
Apply current pulse of appropriate sign to flush the stored flux (the pulse here flushes out positive flux)

- To flush either polarity → Do both ( $\pm$ ) resets in succession

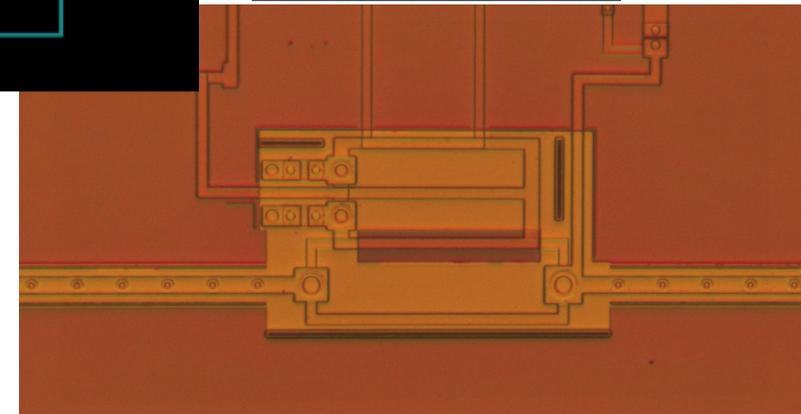
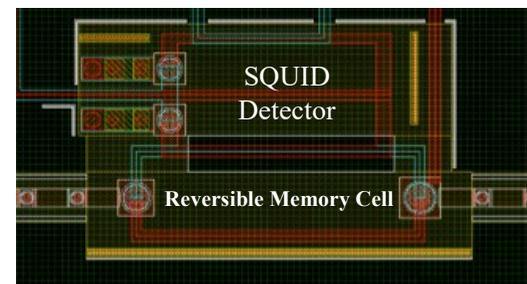


Fabrication at SeeQC with support from ACI

DC-SFQ & LJJ



RM Cell & SQUID



# barc tool for enumerating/classifying BARCS device functions



Custom Python program with 16 modules.

Tool is now complete; will be open-sourced.

Layer-cake view of software architecture:

- Modules only import modules from lower-numbered layers.

```
Symmetry group #38 has 6 functions:
```

```
Function #155.
Function #340.
Function #481.
Function #285.
Function #365.
Function #185.
```

```
Example: Function #155 = [1]*3(L,R):
```

```
1(L) -> (R)2
1(R) -> (L)3
2(L) -> (R)1
2(R) -> (R)3
3(L) -> (L)2
3(R) -> (L)1
```

```
Function #155 has the following symmetry properties:
```

```
It is D-dual to function #481
It is S-dual to function #481
It is E(1,2)-dual to function #340
It is E(1,3)-dual to function #185
It is E(2,3)-dual to function #481
It R(-1)-transforms to function #365
It R(1)-transforms to function #285
```

Layer	Module Names & Descriptions
4	<b>barc</b> (top-level program)
3	<b>deviceType</b> – Classification of devices with given dimensions.
2	<b>deviceFunction</b> – Device with a specific transition function. <b>stateSet</b> – Identifies a set of accessible device states.
1	<b>pulseAlphabet</b> – Sets of pulse types. <b>pulseType</b> – Identifies a specific type of pulse. <b>state</b> – Identifies an internal state of a device. <b>symmetryGroup</b> – Equivalence class of device functions. <b>transitionFunction</b> – Bijective map, input→output syndromes.
0	<b>characterClass</b> – Defines a type of signal characters. <b>deviceDimensions</b> – Defines size parameters of devices. <b>dictPermuter</b> – Used to enumerate transition functions. <b>signalCharacter</b> – Identifies I/O event type (pulse type & port). <b>symmetryTransform</b> – Invertibly transforms a device function. <b>syndrome</b> – An initial or final condition for a device transition. <b>utilities</b> – Defines some low-level utility functions.

← Example description of a symmetry-equivalence group as output by the **barc** tool.

# Symmetry Relations of Interest



The following symmetry relations on BARC functions are considered in this work:

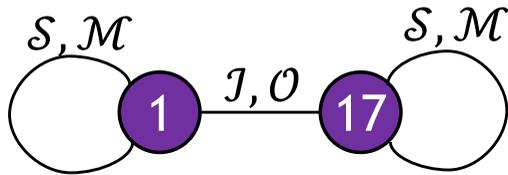
- ***Direction-reversal symmetry***  $\mathcal{D}$  – Symmetry under exchange of input & output syndromes (involution of transition func.)
- ***State-exchange symmetry***  $\mathcal{S}$  – Symmetry under an exchange of state labels (and fluxes, for flux-polarized states).
- ***Flux-negation symmetry***  $\mathcal{F}$  – Symmetry under negation of all (I/O flux & internal state) flux polarities.
- ***Moving-flux negation symmetry***  $\mathcal{M}$  – Symmetry under negation of all moving (I/O) flux polarities.
  - *Input flux negation symmetry*  $\mathcal{I}$  – Symmetry under negation of all input flux polarities.
  - *Output flux negation symmetry*  $\mathcal{O}$  – Symmetry under negation of all output flux polarities.
- ***Port-relabeling symmetries***  $\mathcal{R}_P$  – Symmetry under a particular permutation  $P$  of the port labels.
  - *Port exchange symmetry*  $\mathcal{E}(p_i, p_j)$  – Symmetry wrt an exchange of labels between a particular pair of ports.
  - *Rotational symmetry*  $\mathcal{R}_r$  – Relevant for  $n \geq 3$  ports. Symmetry under (planar) rotation of port labels.
  - *Reflection across port axis*  $\mathcal{R}_{\{p_i\}}$  – Symmetry under reflection of ports on either side of port  $p_i$ .
  - *Mirror symmetry*  $\mathcal{M}_2, \mathcal{M}_3$  – Symmetry under port exchange for a 2-port device, or any reflection for a rotationally symmetric 3-port device.
  - *Complete port symmetry*  $\mathcal{R}(n)$  – Symmetry under *all* possible relabelings of the ports.

# Equivalence Groups For the 24 One-Port, Two-State Elements:

$2 \cdot 1 \cdot 2 = 4$  I/O syndromes  $\rightarrow 4! = 24$  permutations (raw reversible transition functions).

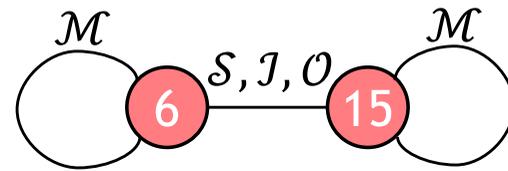


Stateful Reflector



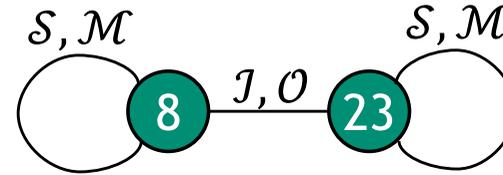
(State Unused—Not Atomic)

Configurable Inverter



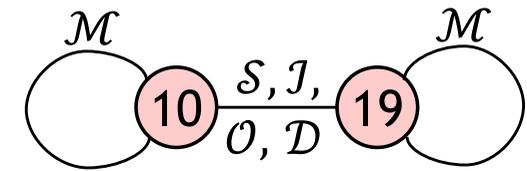
(Doesn't Change State)

Toggle



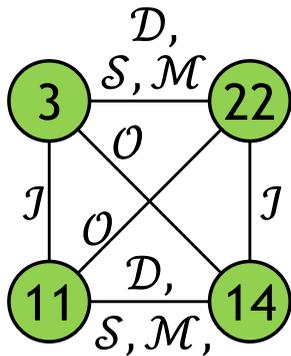
(Doesn't Use State)

Toggle & Conditional Invert

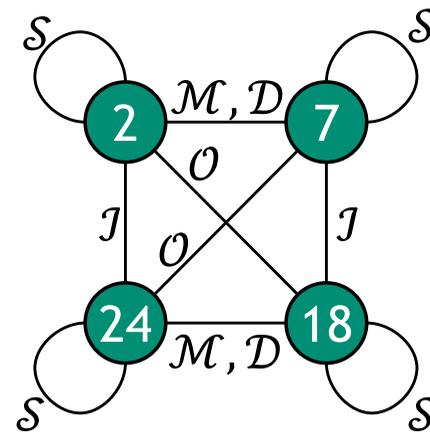


(Neither flux-negation symmetric nor flux-conserving)

Exchange (RM)

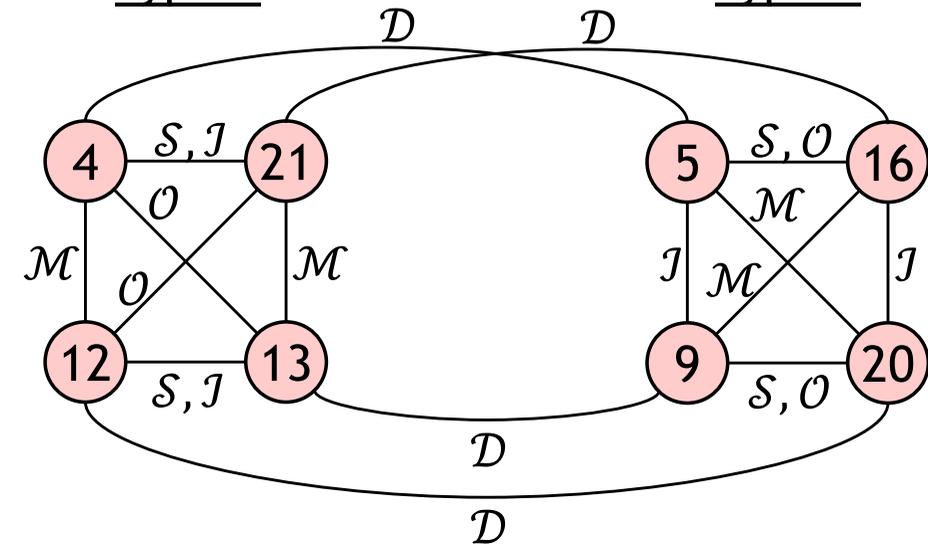


Conditional Toggle



(Doesn't Use State)

Type 4



(Neither flux-negation symmetric nor flux-conserving)

# Two-Port, Two-State, Flux-Polarized Elements

There are  $2^3 = 8$  I/O syndromes, thus  $8! = 40,320$  raw reversible transition functions.

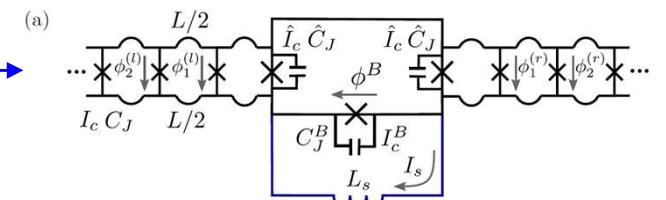
- But only 96 of them satisfy the flux conservation constraint.
- And only 10 of these are nontrivial primitives satisfying all constraints.

These 10 functions sort into 7 equivalence groups as follows:

Self-Symmetry Group Size	Equivalence Group Size	Number of Equiv. Groups	Total # of Raw Trans. Funcs.
4	1	4	4
2	2	3	6
<b>TOTALS:</b>		<b>7</b>	<b>10</b>

The corresponding functional behaviors can be described as:

1. Reversible Shift Register (RSR) – More on this one later.
2. Directed Reversible Shift Register (DRSR)
3. Filtering RM Cell (FRM)
4. Directed Filtering RM Cell (DFRM).
5. Polarized Flipping Diode (PFD). – Also has a flux-neutral equivalent.
6. Asymmetric Polarity Filter (APF).
7. Two-Port Reversible Memory Cell (RM2). – Implemented.



(Osborn & Wustmann '22)

# Illustrations of 2-port, 2-state, flux-polarized elements:

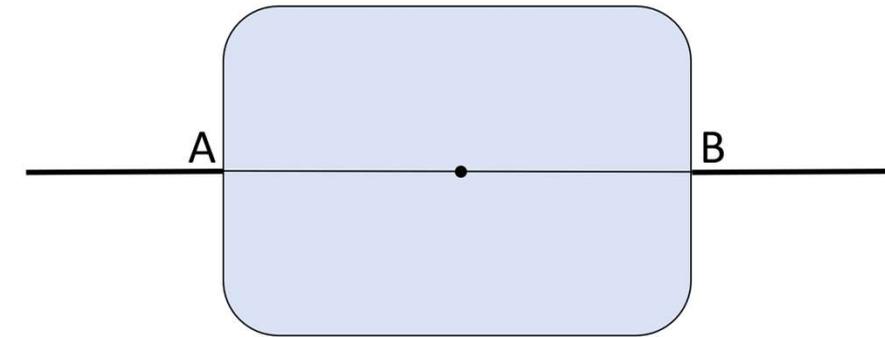


(Table Rows Shown for  $\uparrow$  Initial State Only)

## 1. Reversible Shift Register (RSR):

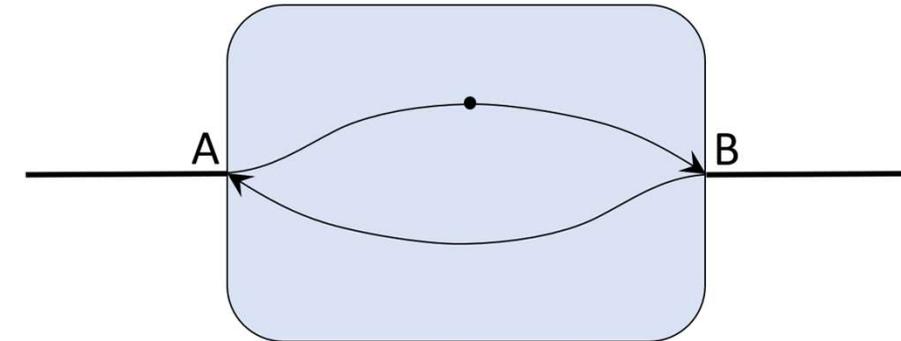
(Implemented by Osborn & Wustmann '22)

Input syndrome	Output syndrome
$\uparrow \rangle A(\uparrow)$	$(\uparrow) B \rangle \uparrow$
$\downarrow \rangle A(\uparrow)$	$(\downarrow) B \rangle \uparrow$
$\uparrow \rangle B(\uparrow)$	$(\uparrow) A \rangle \uparrow$
$\downarrow \rangle B(\uparrow)$	$(\downarrow) A \rangle \uparrow$



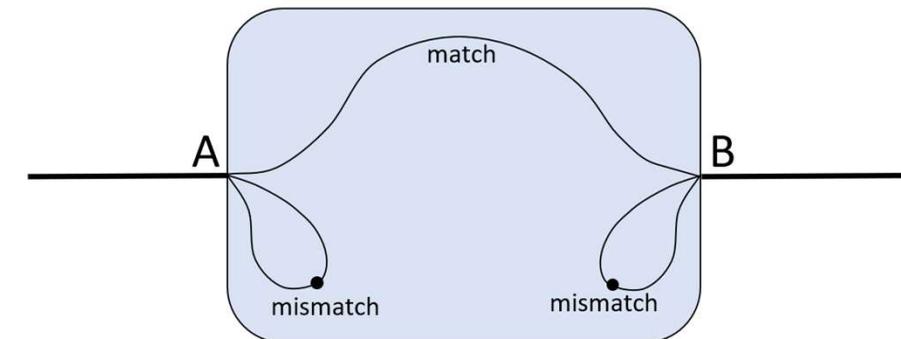
## 2. Directed Reversible Shift Register (DRSR):

Input syndrome	Output syndrome
$\uparrow \rangle A(\uparrow)$	$(\uparrow) B \rangle \uparrow$
$\downarrow \rangle A(\uparrow)$	$(\downarrow) B \rangle \uparrow$
$\uparrow \rangle B(\uparrow)$	$(\uparrow) A \rangle \uparrow$
$\downarrow \rangle B(\uparrow)$	$(\uparrow) A \rangle \downarrow$



## 3. Filtering RM Cell (FRM):

Input syndrome	Output syndrome
$\uparrow \rangle A(\uparrow)$	$(\uparrow) B \rangle \uparrow$
$\downarrow \rangle A(\uparrow)$	$(\downarrow) A \rangle \uparrow$
$\uparrow \rangle B(\uparrow)$	$(\uparrow) A \rangle \uparrow$
$\downarrow \rangle B(\uparrow)$	$(\downarrow) B \rangle \uparrow$



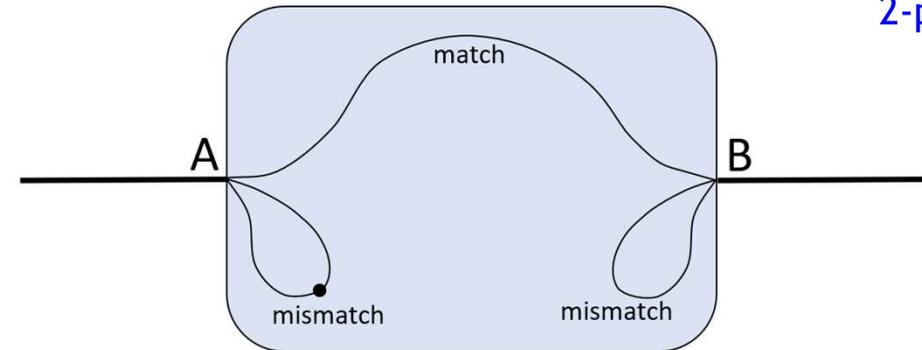
# Illustrations of 2-port, 2-state, flux-polarized elements, cont.:



(Not shown:  
2-port RM cell)

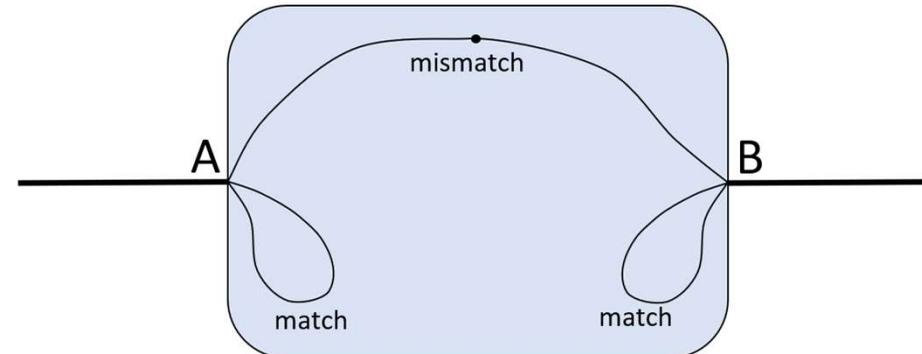
## 4. Directed Filtering RM Cell (DFRM):

Input syndrome	Output syndrome
$\uparrow\rangle A(\uparrow)$	$(\uparrow)B\rangle \uparrow$
$\downarrow\rangle A(\uparrow)$	$(\downarrow)A\rangle \uparrow$
$\uparrow\rangle B(\uparrow)$	$(\uparrow)A\rangle \uparrow$
$\downarrow\rangle B(\uparrow)$	$(\uparrow)B\rangle \downarrow$



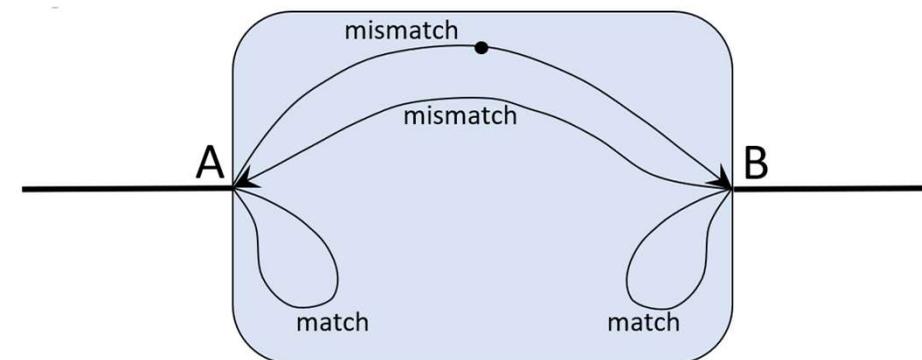
## 5. Polarized Flipping Diode (PFD):

Input syndrome	Output syndrome
$\uparrow\rangle A(\uparrow)$	$(\uparrow)A\rangle \uparrow$
$\downarrow\rangle A(\uparrow)$	$(\downarrow)B\rangle \uparrow$
$\uparrow\rangle B(\uparrow)$	$(\uparrow)B\rangle \uparrow$
$\downarrow\rangle B(\uparrow)$	$(\downarrow)A\rangle \uparrow$



## 5. Asymmetric Polarity Filter (APF):

Input syndrome	Output syndrome
$\uparrow\rangle A(\uparrow)$	$(\uparrow)A\rangle \uparrow$
$\downarrow\rangle A(\uparrow)$	$(\downarrow)B\rangle \uparrow$
$\uparrow\rangle B(\uparrow)$	$(\uparrow)B\rangle \uparrow$
$\downarrow\rangle B(\uparrow)$	$(\uparrow)A\rangle \downarrow$



# Two-Port, Two-State, Flux-Neutral Elements

There are  $(2^2)! = 24$  raw flux-symmetric transition functions.

- 14 of these are nontrivial, atomic functional primitives.

These sort into 4 equivalence groups as follows:

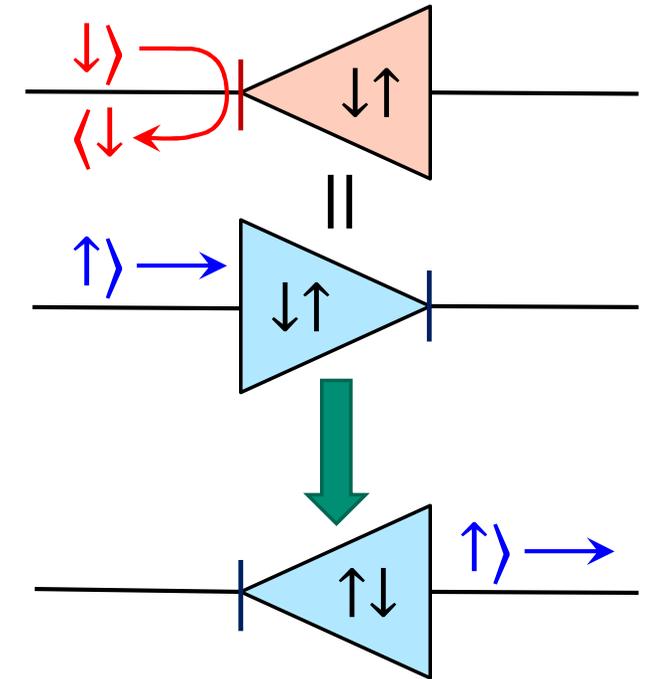
Self-Symmetry Group Size	Equivalence Group Size	Number of Equiv. Groups	Total # of Raw Trans. Funcs.
4	2	3	6
1	8	1	8
<b>TOTALS:</b>		<b>4</b>	<b>14</b>

There are 5 distinct functional behaviors (described in forwards time direction):

1. **Alternating Barrier (AB)**, 2 representations – See next slide.
2. **Polarized Flipping Diode (PFD)**, 2 reps. – **Implemented.**
3. **Variant Polarized Flipping Diode (VPFD)**, 2 reps..
4. **Asymmetric Polarized Flipping Diode (APFD)**, 4 reps.,  
(and this one is  $\mathcal{D}$ -dual to:)
5. **Selectable Barrier (SB)**, 4 reps.

Equivalence group of 8 symmetrically equivalence fns.

Ex: Polarized Flipping Diode (PFD)



**Polarity-Dependent Flipping Diode (PFD)**

I/O Syndrome		
Port	State	Fluxon
Left	$\uparrow\downarrow$	$\uparrow$
Left	$\uparrow\downarrow$	$\downarrow$
Left	$\downarrow\uparrow$	$\uparrow$
Left	$\downarrow\uparrow$	$\downarrow$
Right	$\uparrow\downarrow$	$\uparrow$
Right	$\uparrow\downarrow$	$\downarrow$
Right	$\downarrow\uparrow$	$\uparrow$
Right	$\downarrow\uparrow$	$\downarrow$

# Ex. 2-port, 2-state neutral element: Alternating Barrier (AB)



Flux-conserving, flux-negation symmetric element.

- Also has mirror ( $\mathcal{M}_2$ ) symmetry.
- Has two  $\mathcal{D}, \mathcal{S}$ -dual representations.

Flux-neutral internal states  $\rightarrow$  Doesn't change fluxon polarity.

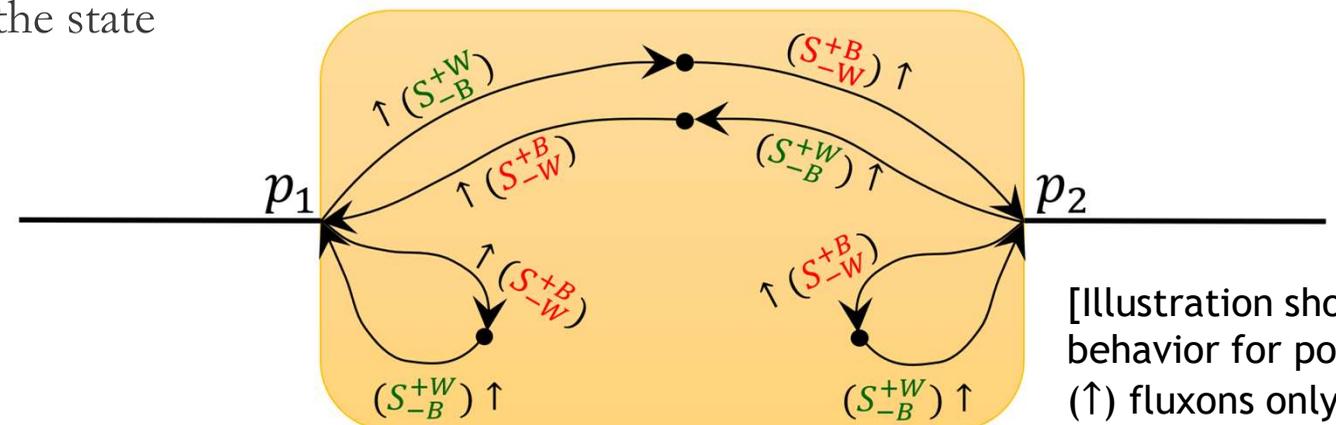
State descriptions:

- $S_{-B}^{+W}$ : *Positive-wire, negative-barrier.*
  - Transmits positive ( $\uparrow$ ) fluxons, reflects negative ( $\downarrow$ ) fluxons.
- $S_{-W}^{+B}$ : *Positive-barrier, negative-wire.*
  - Reflects positive ( $\uparrow$ ) fluxons, transmits negative ( $\downarrow$ ) fluxons.

Transition function description:

- Fluxons arriving at either port are routed as per the state descriptions above.
- State toggles with every interaction.

Input Syndrome	Output Syndrome
$\uparrow\rangle p_1 (S_{-B}^{+W})$	$(S_{-W}^{+B}) p_2 \rangle \uparrow$
$\downarrow\rangle p_1 (S_{-B}^{+W})$	$(S_{-W}^{+B}) p_1 \rangle \downarrow$
$\uparrow\rangle p_2 (S_{-B}^{+W})$	$(S_{-W}^{+B}) p_1 \rangle \uparrow$
$\downarrow\rangle p_2 (S_{-B}^{+W})$	$(S_{-W}^{+B}) p_2 \rangle \downarrow$
$\uparrow\rangle p_1 (S_{-W}^{+B})$	$(S_{-B}^{+W}) p_1 \rangle \uparrow$
$\downarrow\rangle p_1 (S_{-W}^{+B})$	$(S_{-B}^{+W}) p_2 \rangle \downarrow$
$\uparrow\rangle p_2 (S_{-W}^{+B})$	$(S_{-B}^{+W}) p_2 \rangle \uparrow$
$\downarrow\rangle p_2 (S_{-W}^{+B})$	$(S_{-B}^{+W}) p_1 \rangle \downarrow$



[Illustration shows AB behavior for positive ( $\uparrow$ ) fluxons only]

# Summary of Results for Three-Port, Two-State Elements:



(Still assuming flux conservation & flux negation symmetry)

Devices with flux-polarized states:

- $2 \cdot 3 \cdot 2 = 12$  I/O syndromes
- $12! = 497,001,600$  raw reversible funcs.
- 25,920 of these are flux-conserving.
- 288 of those are flux-negation symmetric.
- 245 of those are atomic (primitives).
- 219 of those use the state non-trivially.
- Sort into 39 equiv. groups as follows →

Devices with flux-neutral states:

- $1 \cdot 3 \cdot 2 = 6$  I/O syndromes (for ↑ inputs)
- $6! = 720$  permutations.
- 653 of them are atomic primitives.
- 600 of those use the state non-trivially.
- Sort into 45 equiv. groups as follows:

Summary of (3,2) flux-polarized behaviors

<b>Equivalence Class Size:</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>6</b>	<b>12</b>	<b>Tot.</b>
<b>Self-Symmetry Group Size:</b>	<b>12</b>	<b>6</b>	<b>4</b>	<b>2</b>	<b>1</b>	
No. of Equivalence Classes:	1	4	6	24	4	<b>39</b>
Total number of Functions:	1	8	18	144	48	<b>219</b>

Summary of (3,2) flux-neutral behaviors

<b>Equivalence Class Size:</b>	<b>2</b>	<b>4</b>	<b>6</b>	<b>12</b>	<b>24</b>	<b>Tot.</b>
<b>Self-Symmetry Group Size:</b>	<b>12</b>	<b>6</b>	<b>4</b>	<b>2</b>	<b>1</b>	
No. of Equivalence Classes:	1	1	9	23	11	<b>45</b>
Total number of Functions:	2	4	54	276	264	<b>600</b>

# Illustrations of some 3-port, 2-state flux-neutral elements

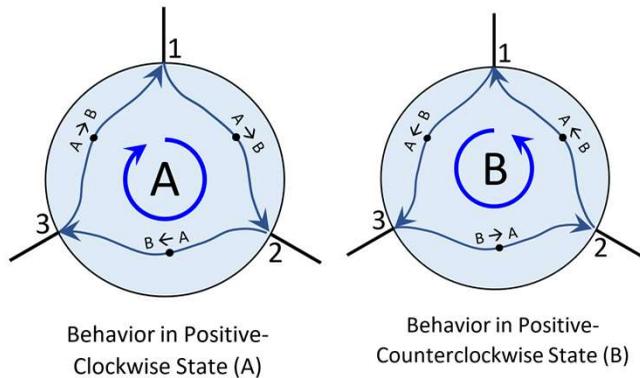


Recall there are 45 different non-trivial, atomic functional behaviors (counting  $\mathcal{D}$ -duals as equivalent).

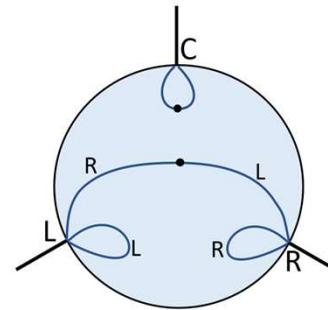
Of these, only a few exemplar behaviors are illustrated here.

Still seeking implementations for any of these....

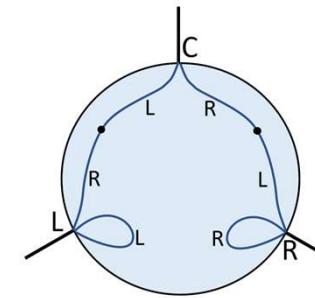
## Polarized Neutral Toggle Rotary (PNTR)



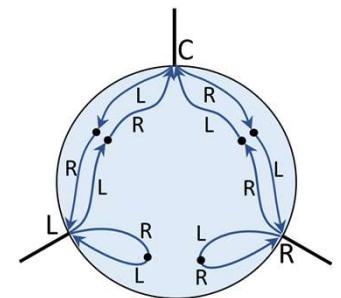
## Polarized Controlled Flipping Diode (PCFD)



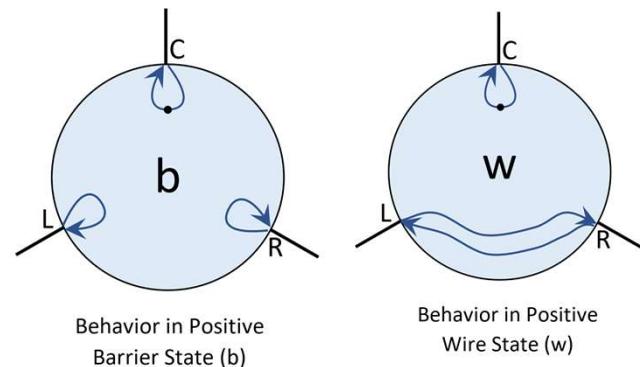
## Polarized Throw Switch, Type A (PTSA)



## Polarized Throw Switch, Type B (PTSB)

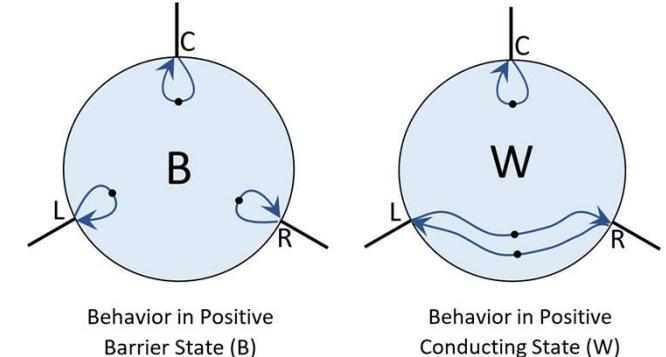


## Polarized Toggle Controlled Barrier (PTCB)



*[NOTE: All behaviors shown here are for (+) fluxons only; (-) fluxons interact oppositely with states]*

## Polarized Knock-twice Toggle Controlled Barrier (PKTCB)



# Example Use of Toggle Rotary (TR): Pulse Duplicator (PD)

A very useful 2-state, 3-terminal element is the *Toggle Rotary* (TR).

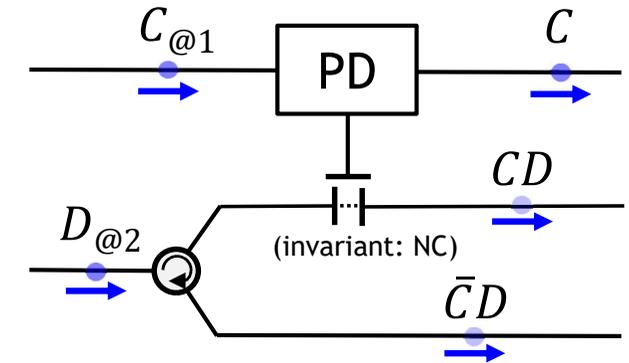
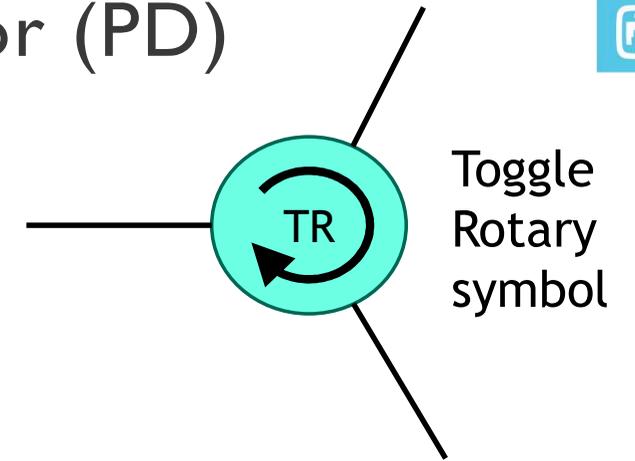
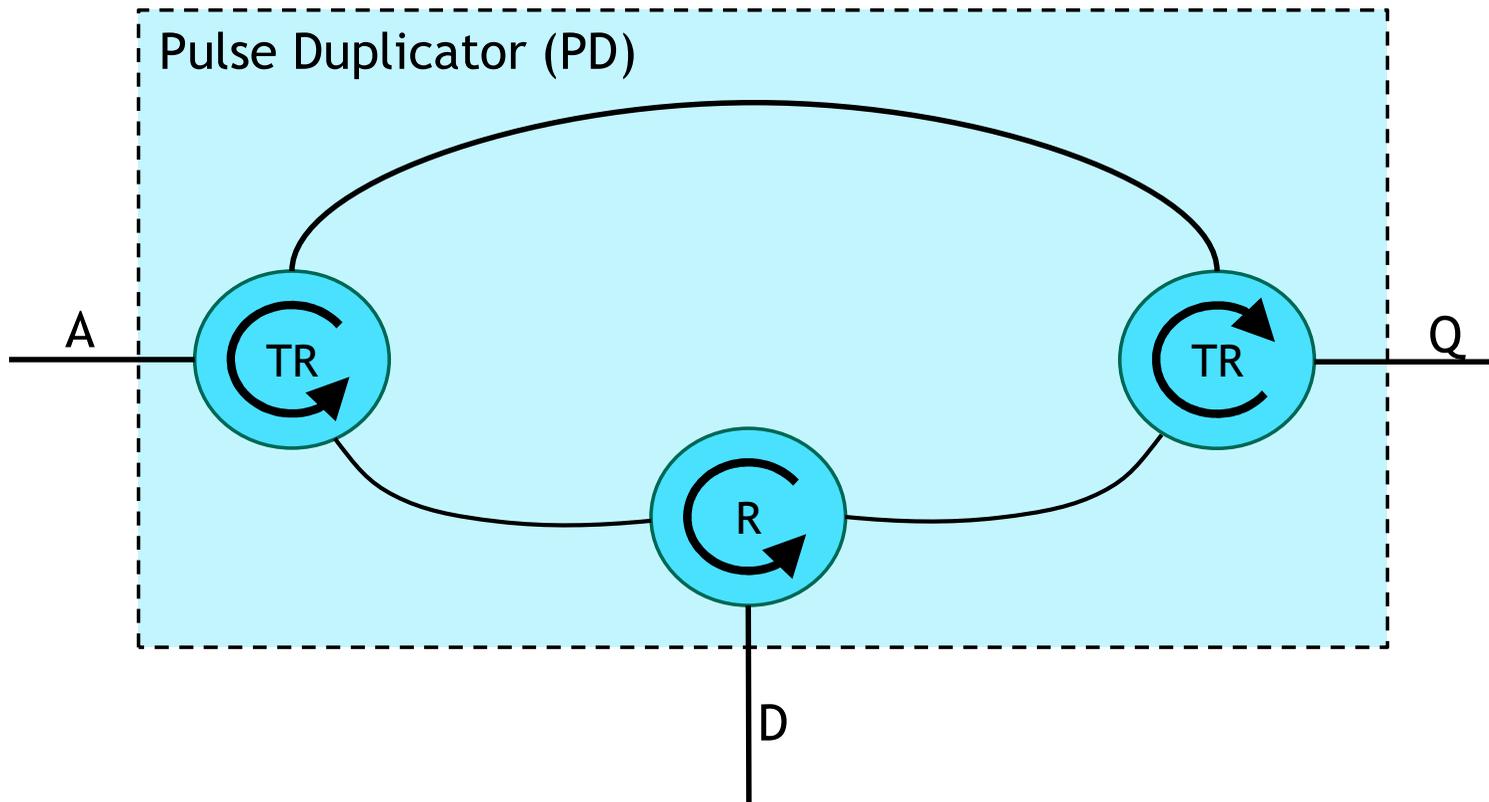
- Toggles its directionality (parity) each time a pulse is routed.
- Comes in unconditional (UTR) & polarized (PTR) versions. (Don't care in unipolar circuits.)

Can be used to construct a (very simple) *Pulse Duplicator* (PD) element.

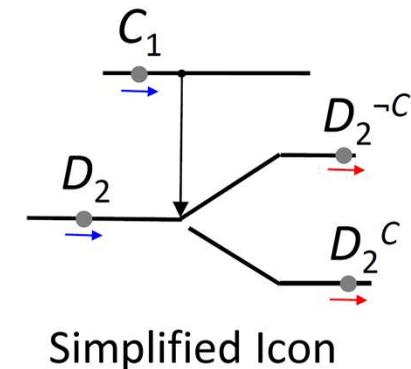
- A pulse arriving at **A** comes out **D** *twice* (if reflected back in between) before emerging at **Q**.

The PD can be used to vastly simplify our original (unipolar) universal circuit construction!

- No more constant streams of input pulses needed for AND. (Still needed for NOT.)



Non-Toggling (Ressler-Feynman equiv.) Switch Gate



Simplified Icon

# Polarity Filter (PF) and Polarity Separator (PS) functions

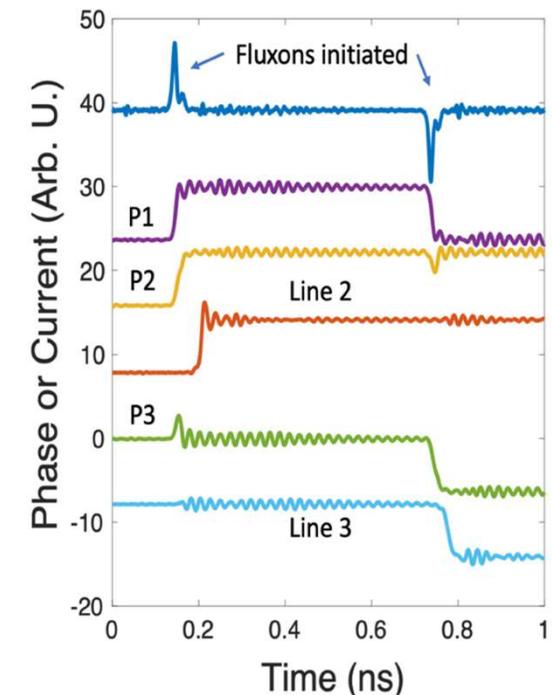
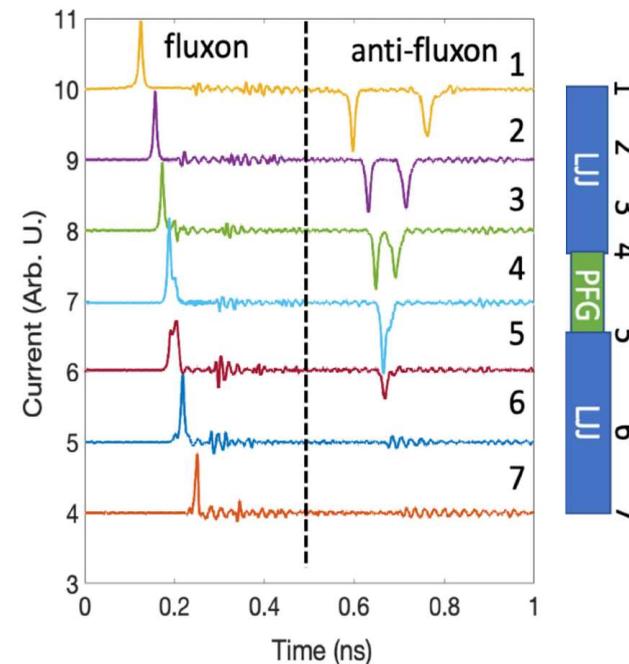
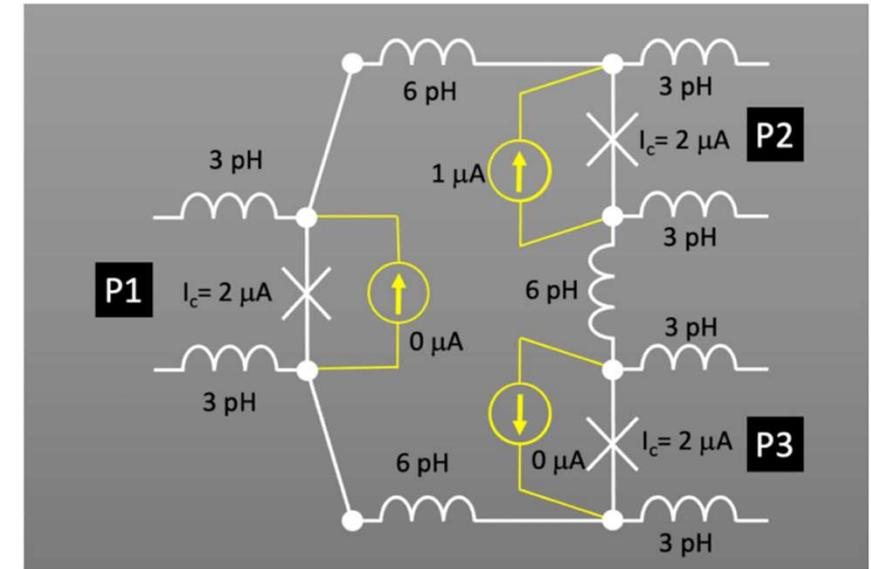
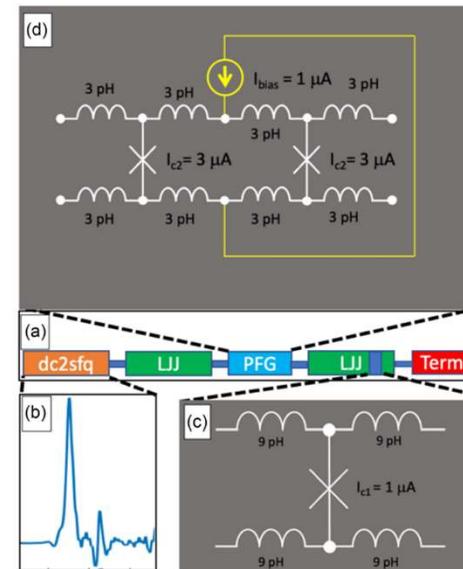
Lewis & Frank, ASC '22, [10.1109/TASC.2023.3244115](https://doi.org/10.1109/TASC.2023.3244115)

Two *irreversible* functions that are still useful in constructing test circuits for BARCS elements.

- 2-port polarity filter (PF) element passes one polarity of fluxon and reflects the other polarity.
- 3-port polarity separator (PS) element routes one polarity of fluxon to one output port, the other to the other output port.

Both of these functions can be configured as desired via the appropriate setting of bias currents.

- In particular, they can be configured for use with input fluxons impinging on any of their ports.



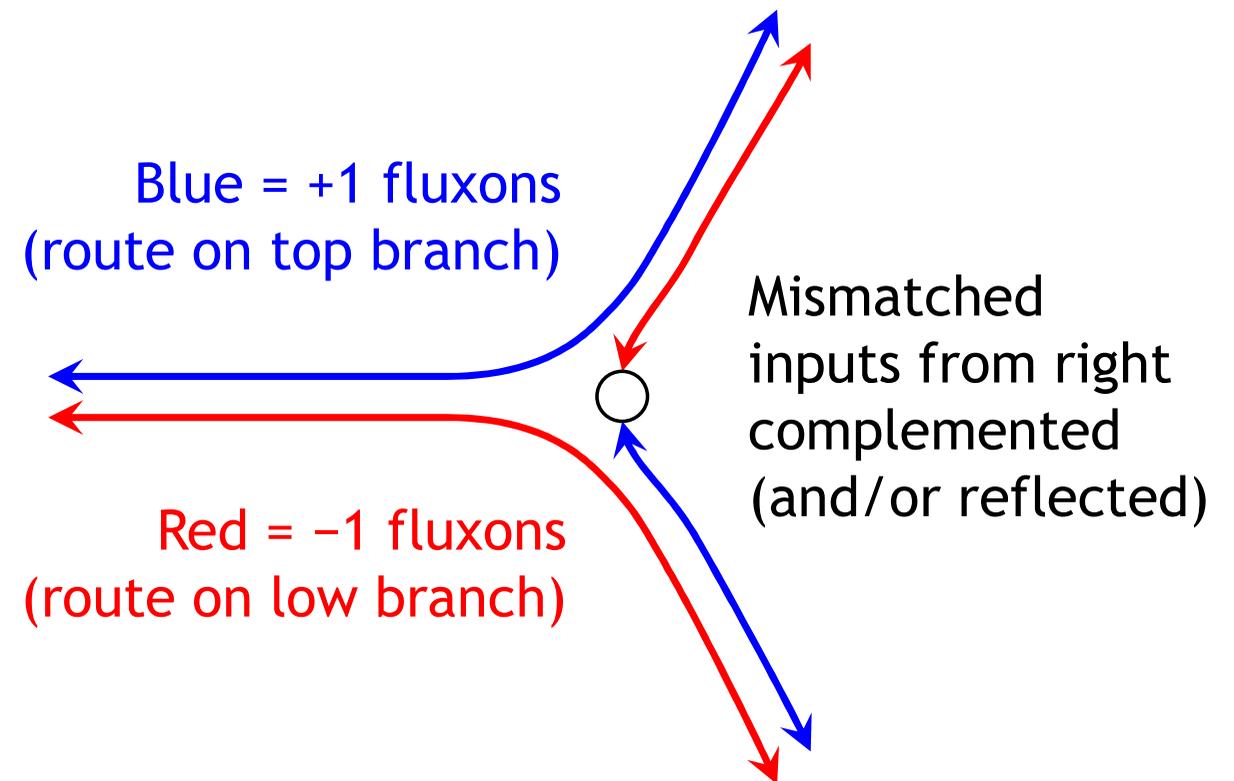
# Reversible Polarity Separator (RPS)

There are several fully-reversible 3-port functional behaviors any of which would constitute a logically reversible variant of the polarity separator...

- The diagram at right illustrates a few cases...

We don't yet know if any such behavior is implementable as a ballistic JJ circuit

- But of course, if there exists any ballistically implementable universal set of BARCS elements, then it follows that these behaviors are implementable...



# LJJ modeling for the MITLL SFQ5ee process



Model based on a rough InductEx analysis of the LJJ stack.

- For a baseline  $1\mu\text{m}$ -wide continuous LJJ structure,
  - Inductance per square was estimated to be  $\sim 0.16\text{ pH}$ .
  - Josephson penetration length was estimated to be  $\sim 4.6\mu\text{m}$ .

A suitable lumped-element model for a continuous LJJ was chosen based on selecting a unit cell length  $a \leq \lambda_J/3$ .

- A  $1\mu\text{m} \times 1\mu\text{m}$  unit cell with  $I_c = 100\mu\text{A}$  is more than small enough for our present purposes, and is conveniently sized.

```

Magnetic flux quantum Phi0      = 2.068fWb
Width of standard LJJ           = 1.000um
JJ critical current density      = 100.0MA/m^2
LJJ inductance per square       = 158.1fH
Josephson penetration depth     = 4.562um
LJJ unit cell length            = 1.000um
LJJ unit cell area              = 1.000p(m^2)
LJJ unit cell critical current  = 100.0uA
LJJ unit cell capacitance       = 70.00fF
LJJ unit cell subgap resistance = 160.0Ω
LJJ unit cell normal resistance = 16.00Ω
LJJ inductance piece            = 39.52fH
  
```

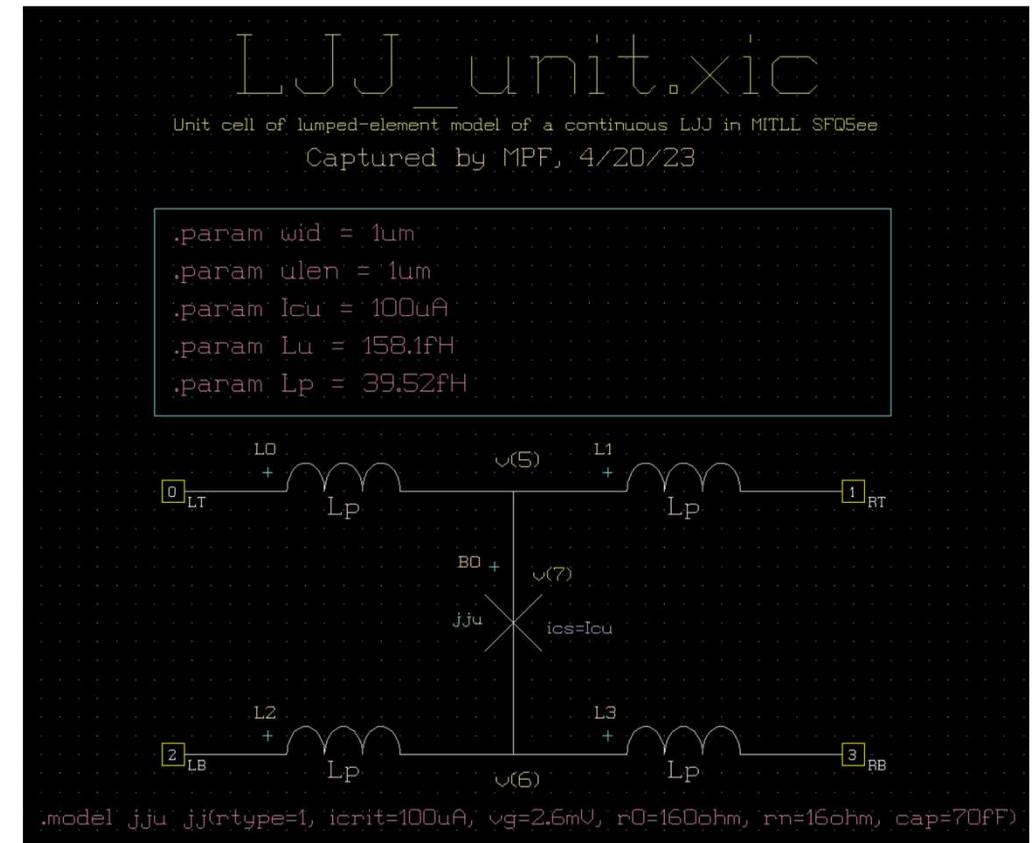
SUBCIRCUIT FOR UNIT CELL:

```

.subckt LJJ_unitcell LT LB RT RB
B0 top bot ph jju ics=100.0uA
L0 LT top 39.52fH
L1 top RT 39.52fH
L2 LB bot 39.52fH
L2 bot RB 39.52fH
.ends LJJ_unitcell
.model jju jj(icrit=100.0uA, r0=160.0ohm, rn=16.00ohm, cap=70.00fF)
  
```

JJ model parameters  
from MIT Lincoln Lab

```
.model JJ1 jj (area=1, rn=16, rsg=160, cap=0.07p, vg=2.6m)
```



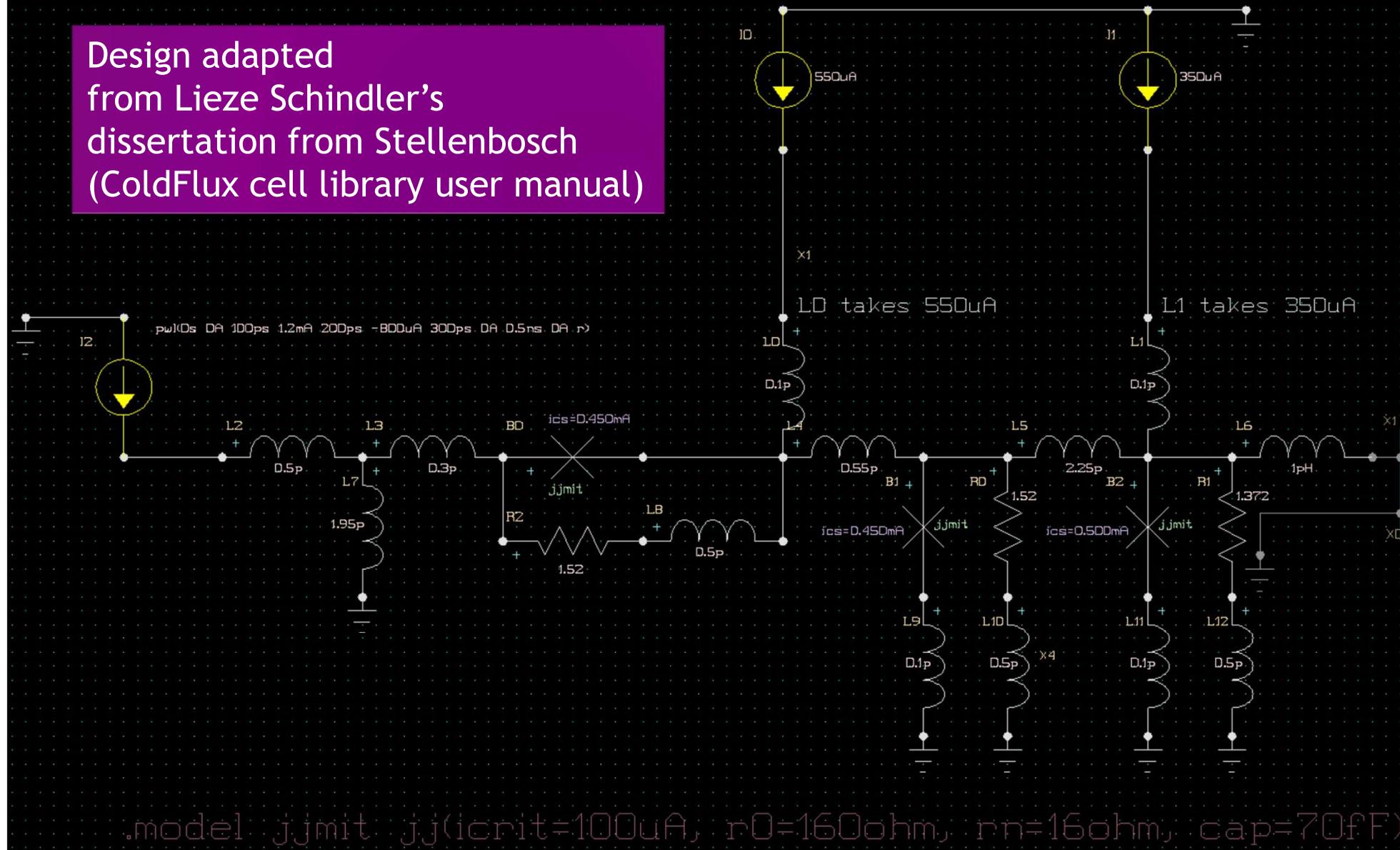
Xic cells for  
prototyping  
test setups:



# DC/SFQ Converter Design for use with our SFQ5ee LJJs



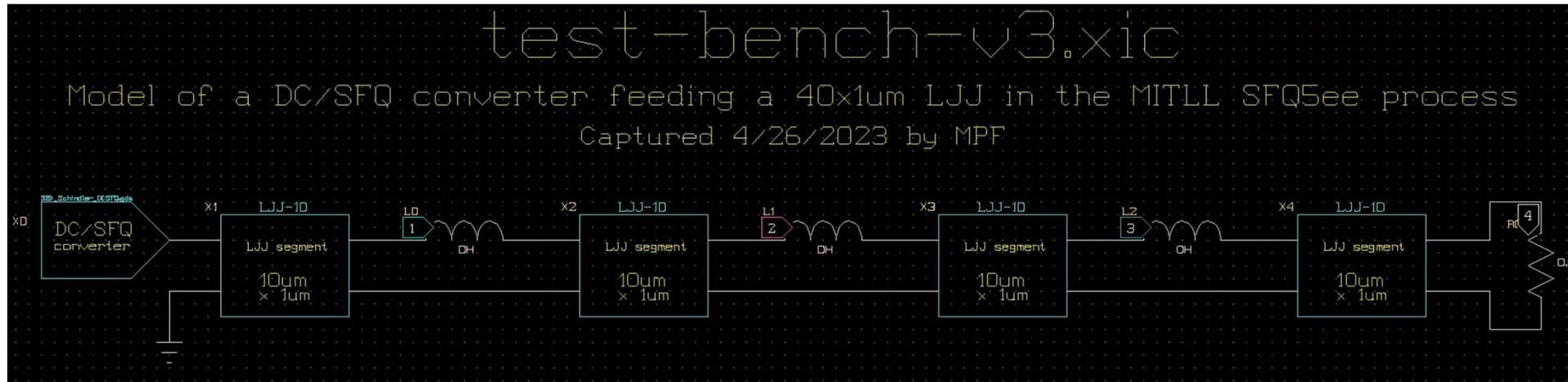
Design adapted  
from Lieze Schindler's  
dissertation from Stellenbosch  
(ColdFlux cell library user manual)



# Characterizing Fluxons on the SFQ5ee continuous LJJ

For this particular combination of DC/SFQ converter and LJJ, the output fluxons:

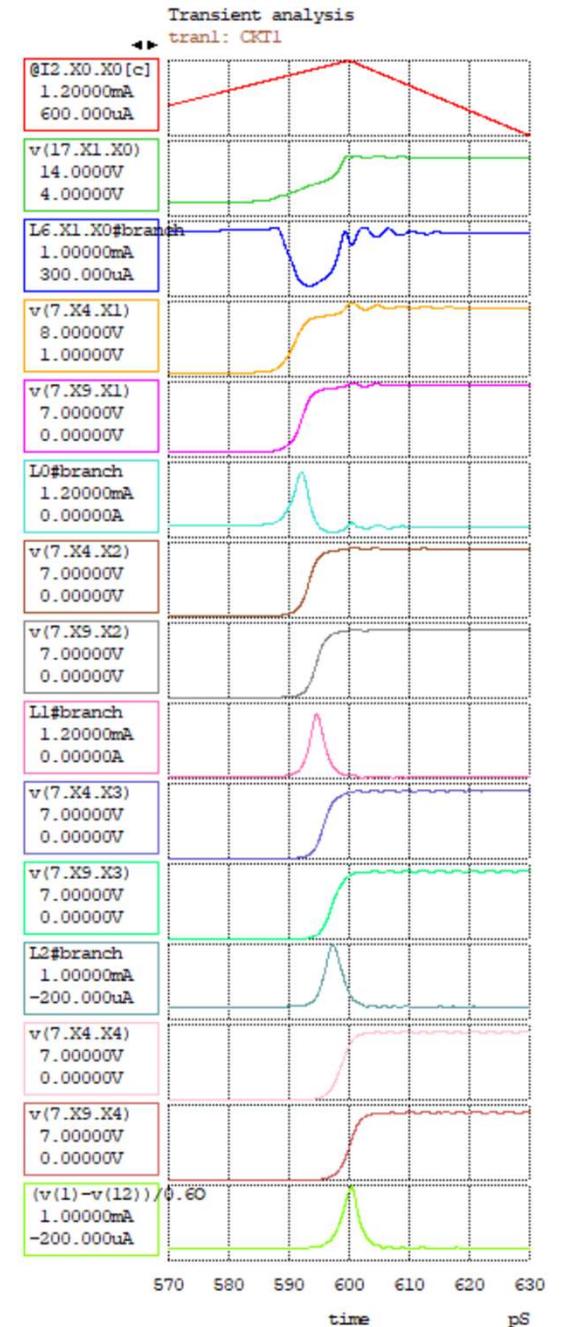
- Have an initial velocity of approx.  $(10 \mu\text{m}) / (2.46 \text{ ps}) = 1.36\%$  of the speed of light.
- Most pulse energy falls in a  $\sim 4 \text{ ps}$  time window  $\rightarrow$  dom.  $\omega \approx 785 \times 10^9 \text{ rad/s.} = 125 \text{ GHz}$
- Calculated line impedance  $Z_{\text{LJJ}} \approx 0.6 \Omega$ ; terminating with this resistance performs well.
- Pulse duration is  $\sim 7 \text{ ps}$ ; spatial extent of pulse is thus  $\sim 28.6 \mu\text{m} \approx 2\pi \lambda_j$  (as expected).
- Swihart velocity  $c_s$  estimated at  $3.17\%$  of lightspeed  $\rightarrow$  initial fluxon velocity is  $\sim 0.43 c_s$ .



DC/SFQ  
Converter

Model of  $40 \mu\text{m} \times 1 \mu\text{m}$  continuous LJJ

$0.6 \Omega$   
term.  
resis.



# Latest Plan for SCIT Development: AI-Enhanced Approach!



Superconducting Circuit Innovation Tool, first proposed at ISEC '19 ([10.1109/ISEC46533.2019.8990900](https://doi.org/10.1109/ISEC46533.2019.8990900))

1. Gather data from a large number of simulation runs (can parallelize on cluster compute resources)...
  - Monte-Carlo synthesis of random circuit topologies (up to a certain size) & random component values (in certain ranges).
  - For each circuit, subject it to multiple sequences of input symbols (defined by the port & polarization of incoming fluxons), and capture the resulting sequence of output symbols.
    - Greedy strategy to narrow down function. Continue until only 0 or 1 function remains in the target category (defined by #s of ports and states).
  - Record the raw results of the simulation runs, as well as the identity of the remaining function (if any), and its average energy efficiency.
2. Use this dataset to train up a suitable machine learning (ML) model. (Graph neural net, autoencoder, *etc.*) The goal of this would be that, after training, the resulting model will have learned to solve both:
  - The **forward problem**: Given a circuit design, what function does it implement?
    - Or: Given a circuit and an input sequence, what output sequence would it produce?
  - The **inverse problem**: Given a desired function, what is a circuit design that would implement it?
    - Or: Given a desired function and target level of energy efficiency, what is a circuit design that would hit that target?
3. Use the ML model in inverse mode to generate candidate implementations of desired functions.
  - Test these in the simulator to validate; feed results back into the training dataset to improve predictions.
4. Optimize parameters of working circuits & characterize their operating margins.
  - Can use methods like *e.g.* stochastic gradient descent/genetic algorithms for circuit optimization.

# Simulator Setup for Characterizing Arbitrary Circuit Elements

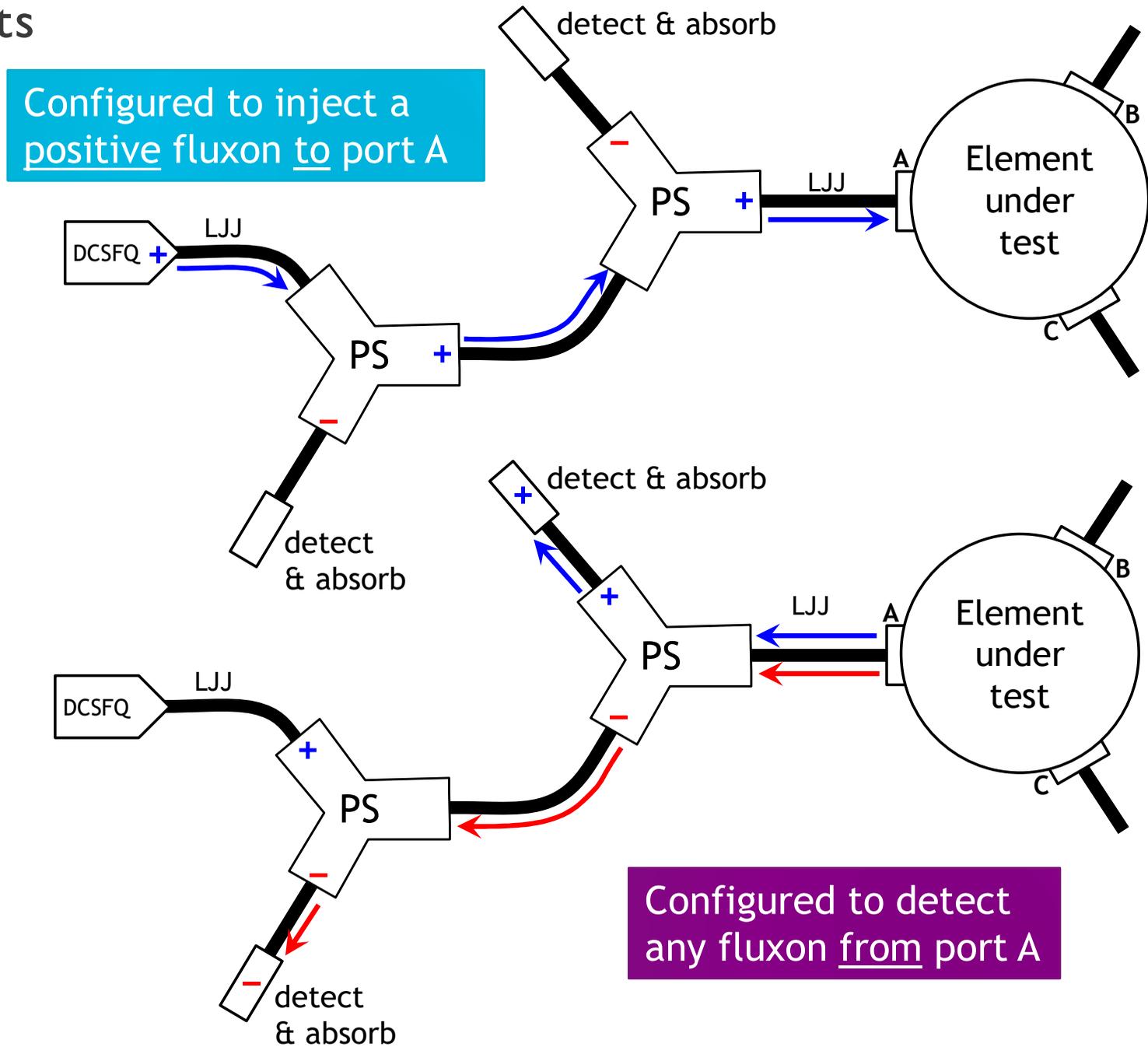
Using polarity separator (PS) elements, we can equip each port of a given circuit element to be tested with a test rig capable of:

- Injecting arbitrary fluxons in to the port,
- Detecting & absorbing arbitrary fluxons coming out of the port.

Each port's rig has three modes; the mode is selected at sim. time by configuring bias currents:

1. Inject positive fluxon.
2. Inject negative fluxon.
3. Detect & absorb either type of outgoing fluxon.

Can switch between modes as quickly as needed in simulations; alternating between inject/detect modes in a test sequence.



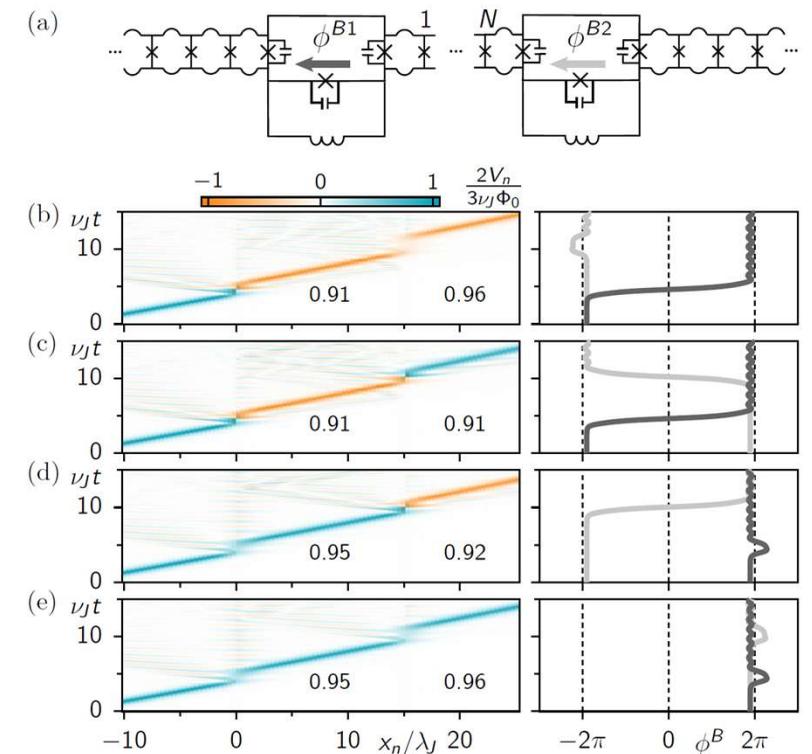
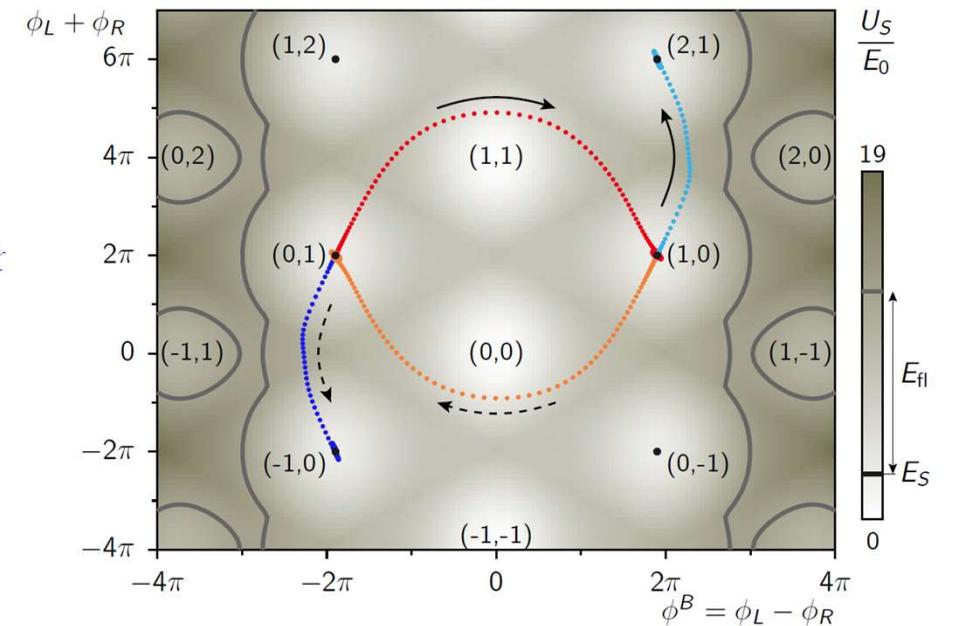
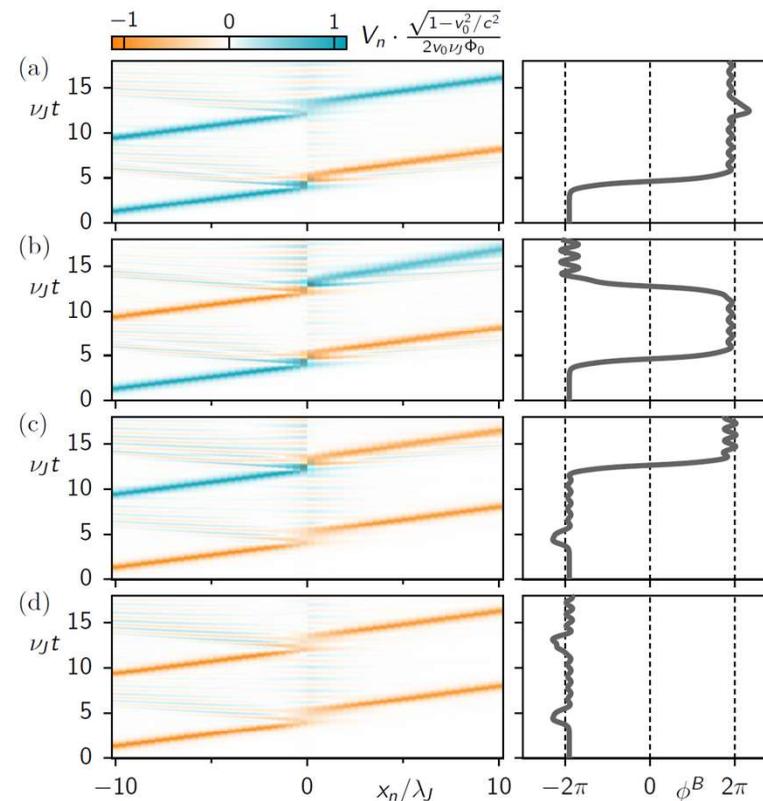
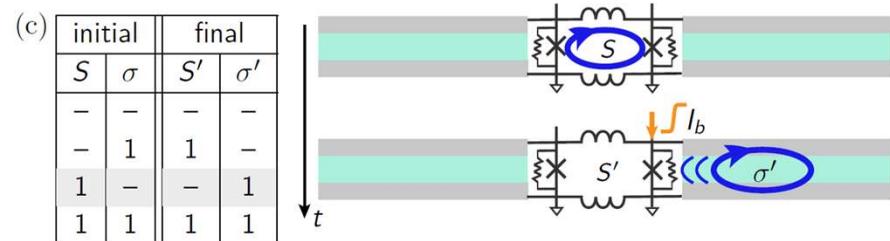
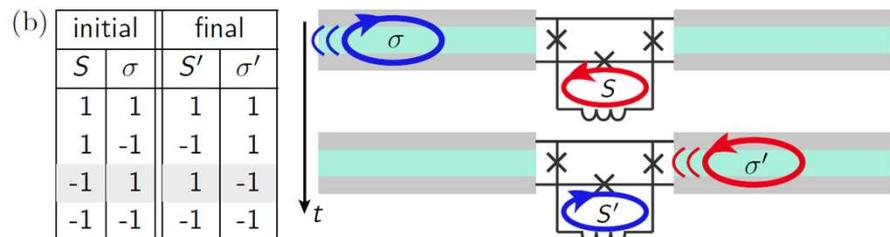
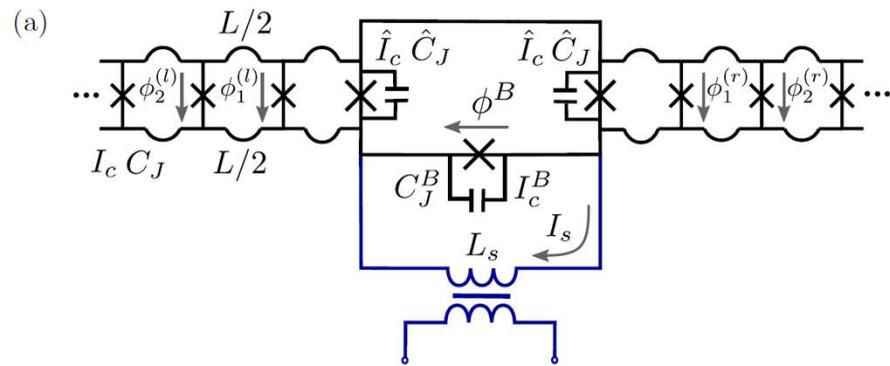
# Ballistic Shift Registers

Work by Osborn & Wustmann, [arxiv:2201.12999](https://arxiv.org/abs/2201.12999)

First working multi-port elements in the BARCS paradigm!

- A type of two-port RM cell in which the output fluxon emerges from the port opposite the input. (Left-right symmetric.)

Work includes detailed simulation & analysis of circuit dynamics.





# Gigahertz Sub-Landauer Momentum Computing

Work by Ray & Crutchfield, [arxiv:2202.07122](https://arxiv.org/abs/2202.07122)

[animation](#)



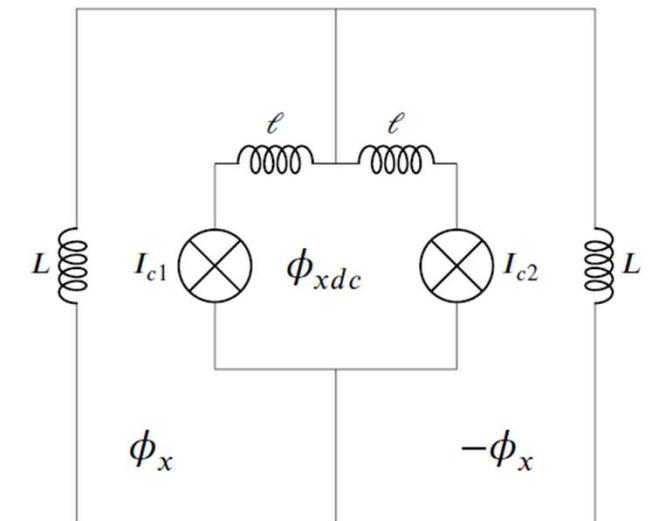
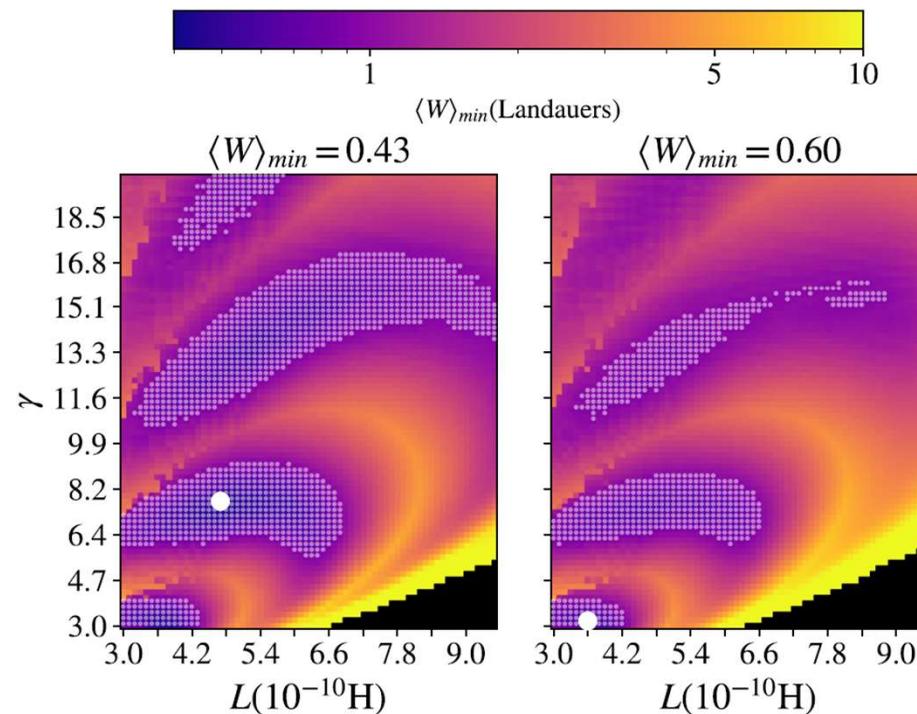
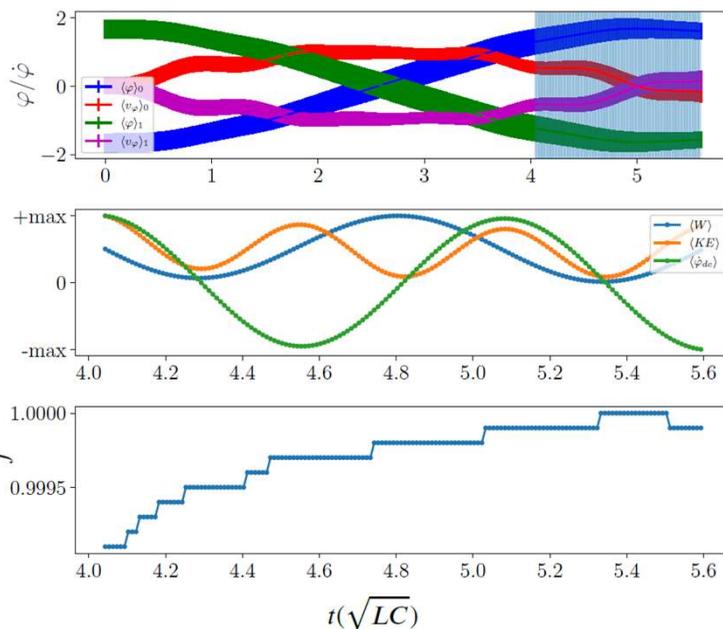
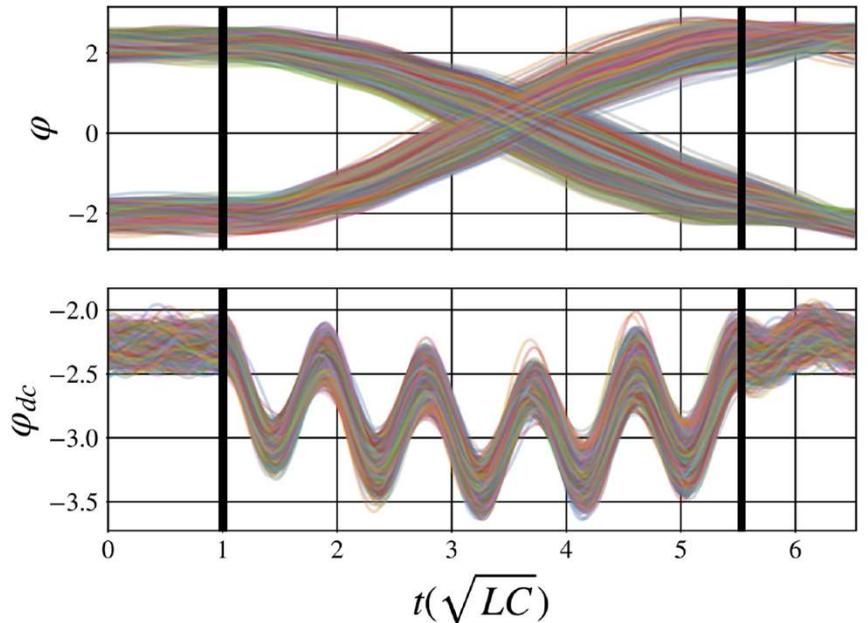
Example protocol implementing a “bit swap” in an SCE circuit.

- State variable corresponds to phase difference between two junctions.
- Somewhat idealized control – instantaneous switching of applied fields.

Temperature-dependent circuit simulation & analysis in a non-equilibrium stochastic thermodynamics framework.

- Accounts for thermal fluctuations of trajectories in the phase space.

Substantial region of parameter space where dissipation is found to be less than  $kT \ln 2$ .



Gradiometric flux logic cell

## Some Next Steps for the BARCS effort



1. Document classification results more fully (in progress).
2. Finish developing **SCIT** (Superconducting Circuit Innovation Tool) tool to facilitate discovery of circuit-level implementations of BARCS functions.
  - Including training an AI/ML model to quickly solve the inverse (circuit design) problem.
3. Better understand role of physical symmetries in the circuit design of BARCS elements.
  - What, if any, functions are ruled out by the symmetries?
  - Must we consider including additional SCE device types to break the symmetries?
4. Identify a computation-universal set of primitive elements that we also know how to implement!
  - Or, show that this is impossible using the present set of devices.
5. Additional work on fabrication & empirical validation of BARCS circuit designs.
6. Gain a better understanding of the limits of the energy efficiency of this approach.

Clearly, much work along these lines remains to be done!

- We would be very happy to recruit new collaborators

## Conclusion



The long-neglected *ballistic* mode of reversible computing has recently attracted renewed interest.

- Classic problems with synchronization & chaotic instability in ballistic computing schemes appear to be resolvable via the asynchronous approach.
- The new method seems to hold some promise for possibly achieving improved energy-delay products and/or more compact circuit designs vs. adiabatic approaches.

Also, note that ballistic approaches are not viable at all in CMOS!

- CMOS has nothing like a ballistic flux soliton, & has no nonlinear reactive elements like JJs...
- Thus, we are leveraging unique advantages of superconducting electronics in this approach.

In this paper & talk, we reported our progress on enumerating & classifying the possible BARCS functions...

- Given constraints of full logical reversibility, flux conservation, & flux negation symmetry.

Multiple US-based research groups in superconductor physics & engineering are now making early progress along this line of work...

- We invite additional domestic & international colleagues to join us in investigating this interesting line of research!